ROUTING IN STOCHASTIC ENVIRONMENTS

A Thesis
Presented to
The Academic Faculty

by

Emrah Uyar

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Industrial and Systems Engineering

Georgia Institute of Technology
December 2008

ROUTING IN STOCHASTIC ENVIRONMENTS

Approved by:

Martin W. P. Savelsbergh, Co-advisor
School of Industrial and Systems Engineering
Georgia Institute of Technology

Alan L. Erera, Co-advisor
School of Industrial and Systems Engineering
Georgia Institute of Technology

Anton J. Kleywegt
School of Industrial and Systems Engineering
Georgia Institute of Technology

Mark Ferguson
College of Management
Georgia Institute of Technology

Ozlem Ergun
School of Industrial and Systems Engineering
Georgia Institute of Technology

Date Approved: 6 November 2008

To my family,

for their unconditional love...

ACKNOWLEDGEMENTS

I would like to thank my advisors Martin Savelsbergh and Alan Erera with all my heart not only for their continuous support and guidance but also for their friendship. I feel privileged to work with them. They have always provided valuable insights and taught me how to tackle issues in life, which I will carry with me forever.

I am deeply thankful to Anton Kleywegt for his support and guidance which made this thesis better. I am also grateful to Ozlem Ergun and Mark Ferguson, who spared their valuable time to serve in my committee, for their insights and valuable comments.

During the four years I spent at Georgia Tech, I was fortunate to meet remarkable people whose friendships mean a lot to me. Okan Orsan Ozener, Ali Ekici, Halil Ozan Gozbasi, Gokhan Samur, Burak Karacik, Aykagan Ak, Serhan Duran, and Evren Ozkaya are a few names I feel obliged to mention. Thank you for the memories.

Finally, I would like to express my deepest gratitude to my parents Mehmet and Gulseren, and my brother Ozkan. Their unconditional love, support, and encouragement have made this possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Vehicle routing problems have been studied extensively in the past decades due, in part, to their practical relevance. Until a few years ago, however, the focus has been on static and deterministic variants, whereas in practice most routing problems are dynamic and stochastic. Dynamic and stochastic vehicle routing problems more closely resemble the situations encountered in practice, but they are harder to analyze and to solve. In this thesis, we design, implement, and analyze new approaches for two dynamic and stochastic vehicle routing problems and provide new insights into how to most effectively handle dynamic and stochastic aspects of routing problems.

In the first two chapters of the thesis, we focus on the following distribution setting. A planner needs to construct a set of delivery routes to be operated daily and to serve a set of geographically dispersed customers. The construction of the set of delivery routes is complicated by the fact that due to the business requirements the same drivers have to visit the same customers as much as possible. This is nontrivial because of the stochastic nature of the demand patterns and the presence of delivery window restrictions. To satisfy the requirement that customers need to be visited by the same driver as much as possible and to achieve cost-efficiency at the same time, we introduce a policy in which each customer can be visited by at most two drivers, i.e., can appear in at most two planned routes. This differs substantially from the approach typically taken when designing fixed routes, but it is appropriate when having to accommodate delivery window restrictions. (We are the first to consider the incorporation of delivery window restrictions when constructing fixed routes.)

In the first chapter of the thesis, we develop heuristic approaches for constructing fixed routes respecting the new policy for large real-life instances. Among the key contributions is the introduction of sampling-based techniques to handle the feasibility issues arising from the delivery window restrictions. An extensive computational study based on real-life

data demonstrates the efficacy of the proposed fixed routing system and route construction techniques.

In the second chapter of the thesis, we investigate the new policy in a more abstract setting to learn more about its properties. In the abstract setting we consider all customer locations are in the unit interval, i.e., [0,1], with the depot at 0. We first study the traditional fixed routes problem and show that optimal fixed routes can be found for certain classes of instances, for example instances in which the customer order probabilities are non-decreasing with the distance to the depot. Next, we present a series of results for the new policy. Among others, we show that the operational feasibility of a set of fixed routes can be established in polynomial time (by solving a maximum cardinality matching problem), but that identifying the optimal operational use of a set of fixed routes for a given demand realization is NP-complete. We also compare various fixed routes structures.

In the third chapter of the thesis, we focus on a distribution setting that arises when there are service level agreements in place between a distributor and its customers. In the specific setting we study, the distributor has to serve customer orders within two days after the order is received, but the distributor has the flexibility to choose the actual delivery day. Because future customer orders are unknown and revealed dynamically over time, deciding the delivery day for orders is nontrivial. The planner tries to minimize total delivery costs over the planning horizon by deciding when to serve each customer by using the probabilistic information regarding future customer orders. We develop heuristic and optimal policies for simple settings of the problem. More specifically, we consider settings in which a single customer order arrives per day and in which customer locations are on the unit interval, the unit circle, and the unit disk. We empirically compare the performance of the various policies with the performance of policies that do not use future information and with an offline optimal policy which has perfect information about future orders. We extend some of the simpler policies to the general setting in which multiple customers arrive each day and in which customer locations are on the Euclidean plane. A computational study shows the value of using probabilistic information regarding future customer orders.

CHAPTER I

INTRODUCTION

1.1 Distribution Management

Many stages can be identified in the transfer of goods from the source of supply to the place of consumption. This dissertation focuses on the last stage: the distribution of finished goods to the end customer. The distribution of finished goods to the end customer typically involves a delivery operation with a central facility (or distribution center), a fleet of vehicles (privately owned or operated by a third party), and a set of geographically dispersed customers (e.g., retail outlets, stores, individuals). Since it is frequently the end customer that initiated the chain of events that led to the finished goods delivery, service related measures, such as on-time delivery, are extremely important at this stage of the supply chain. Supply chain practices, of course, change over time in response to changing economic conditions. As a result, nowadays, more end customers demand smaller quantities of finished goods delivered just in time for consumption. Therefore, distribution operations have become more complex and the associated costs have increased. Balancing service and cost is at the heart of distribution management (and at the heart of supply chain management). Companies have to configure their distribution operations so as to be responsive to the needs of the end customer, but at the same time to maximize utilization of their resources and minimize distribution costs. Distribution costs are under pressure due to the rising fuel costs.

As mentioned above, this dissertation focuses on distribution management or short-haul freight transportation, i.e., the pick-up and/or delivery of goods in a relatively small geographical area. This is an active research area with enormous practical relevance as it is such a common activity in many industries. Generally, a fleet of trucks is used for short-haul freight delivery operations, because trucks are versatile, flexible, and offer fast and reliable service over short hauls. At the heart of much of the research in this area is the Vehicle

Routing Problem (VRP). It is one of the most challenging, yet simply defined, optimization problems, in which a minimum cost set of vehicle routes needs to be constructed to satisfy customer demand while respecting truck capacity limits. Enormous efforts have been expanded on developing effective solution methodologies for the vehicle routing problem and on developing variants that handle additional practical considerations, most notably delivery time windows.

In a sense, we continue that trend, and focus on practical considerations that have become much more relevant in recent years: uncertainty and dynamism. Many of the elements characterizing a real-life routing problem, such as demands and travel times, are stochastic rather than deterministic. Thus incorporating uncertainty into vehicle routing models and solution approaches may lead to more useful and more cost-effective decision support tools. Not only are many of the elements characterizing a real-life routing problem uncertain, information regarding these elements may become available over time rather than all at once. Technological advances in communication and information systems, such as global positioning systems and mobile two-way communication devices have led to more complex and dynamic distribution systems in which information defining the state of the system continuously changes.

This dissertation studies these two particular aspects of routing problems, i.e., stochastic and dynamic information, in two specific settings, both motivated by real-life applications.

1.2 Stochastic and Dynamic Vehicle Routing Problems

As mentioned earlier, the VRP is one of the most studied combinatorial optimization problems due to its practical relevance. The objective is to construct a minimum cost set of vehicle routes visiting every customer exactly once, satisfying the demand of every customer, and doing so without violating vehicle capacity limits. In the VRP, it is assumed that all the necessary information, i.e., travel costs, customer demands, and vehicle capacities, is known in advance of route construction. In many real-life distribution settings, there is some level of uncertainty associated with the relevant information, for instance customer demands may not be known exactly until the customer is visited. In stochastic vehicle routing problems, one or more pieces of the necessary information are stochastic, for example the demands at the customers may not be known exactly, but may be known only in distribution. Stochastic vehicle routing problems are significantly more complex than their deterministic counterparts. Consider the Vehicle Routing Problem with Stochastic Demands (VRPSD). Since the exact demand at a customer is not known at the time vehicle routes are constructed, it may not be possible, or it may be extremely costly, to guarantee that the vehicle capacity will not be violated for any realization of customer demands. Also, since the exact demands at customers are not known at the time vehicle routes are constructed, the objective has to be to construct a set of routes with minimum expected costs. To handle the vehicle capacity issue (or, more generally, feasibility issues) new concepts have to be introduced. The two most popular ones are chance constraints and recovery actions. To avoid extremely costly solutions when capacity feasibility has to be guaranteed for all realizations, a chance constraint requires capacity feasibility only for a fraction of all possible demand realizations, e.g., for 95% of the realizations the constructed set of vehicle routes has to be capacity feasible. Recovery actions specify in advance how to handle capacity infeasibilities if they arise for a particular demand realization, e.g., return to the central facility to reload the vehicle and then resume the route. A cost is incurred for recovery actions and this cost has to be incorporated in the expected costs of a set of routes. It should be clear that solution approaches for stochastic vehicle routing problems are likely to be much more computationally intensive. Fortunately, advances in computer hardware and in optimization algorithms have brought stochastic vehicle routing problems within reach. Note that in stochastic vehicle routing problems the goal is to construct a set of *a priori* routes and that these *a priori* routes will be executed as planned, resorting to pre-specified recovery actions if need be.

Various stochastic vehicle routing problems have been studied in the literature depending on what part of the necessary information is stochastic. We have already introduced the VRPSD. A related but different variant is the Vehicle Routing Problem with Stochastic Customers (VRPSC), where it is not the size of the demand that is stochastic, but it is uncertain whether or not a customer will place an order. Not surprisingly, there is also the Vehicle Routing Problem with Stochastic Customers and Demands (VRPSCD), which integrates the two types of uncertainties into one problem. Another well-known variant is the Vehicle Routing Problem with Stochastic Travel Times (VRPSTT), where the travel time between two locations is stochastic and thus the costs associated with a set of vehicle routes is stochastic. Note that the uncertainty here is only in the cost function and there are no capacity feasibility issues.

In many real-life routing problems there are other complicating factors, most often due to time considerations. For example, there is often a limit on the duration of a vehicle route as a result of driver shift length and/or government regulations. Also, many customers require deliveries to take place during a specific time window. These complications are especially hard to deal with in a stochastic setting because the total travel time of an individual vehicle route is often uncertain (note that the recovery actions typically add travel time to a vehicle route). This is one important contribution of the thesis: we study a VRPSD in which customers have hard delivery windows.

As stated before, in stochastic vehicle routing problems the goal is to construct a set of *a priori* routes and that these *a priori* routes will be executed as planned, resorting to pre-specified recovery actions if need be. Another approach to handling uncertainty in routing problems is to dynamically change the set of routes as more information or more accurate information becomes available. This may be the only option if it is impossible to obtain distribution information. Dynamic vehicle routing problems are those in which the set of vehicle routes is changed dynamically as more or more accurate information becomes available. The prototypical example, which has become known as the Dynamic Vehicle Routing Problem (DVRP), is when the customers that need to be visited and their associated demand are revealed over time during the execution of the vehicle routes, which are continuously updated. The process of updating the routes is often referred to known as "re-optimization" or "re-planning." Typically, it is assumed that the decision maker has no knowledge at all about future customer orders in DVRP and that he uses only the planned vehicle routes and the new customer order information during the re-optimization process.

Many different dynamic vehicle routing settings can be considered and are studied in literature. Environments may differ in the frequency with which the planned vehicle routes can be updated. In some environments, it is possible to update the planned routes every time a new customer order arrives, while in other environments routes can be constructed and updated only at fixed points in time, e.g., once every day or once every hour. Furthermore, environments may differ in the amount of information available at the start of the planning process, e.g., at the start of the planning process 20% of the orders are known or at the start of the planning process 80% of the orders are known. This characteristic is referred to as the degree of dynamism in the system. As a result of the dynamic nature of these routing variants it is more complicated to specify a "solution." It is no longer simply a set of vehicle routes. A "solution" is a policy or an algorithm that decides how to alter the existing set of routes based on any new information. As such simulation is often used to empirically evaluate the performance of dynamic routing policies and algorithms. Another popular measure is the competitive ratio of a policy or algorithm. The competitive ratio tries to capture the value of knowing all the information up front. A more precise definition will be provided in a later chapter. Of course, many variants of dynamic vehicle routing problem can be studied based on which of the complication characteristics of real-life routing problems are considered, such as delivery windows.

Dynamic vehicle routing problems have become more important due to technological advances such as Global Positioning Systems (GPS), mobile two-way communication device, and Electronic Data Interchange (EDI). These technologies allow companies to receive orders at any time, to know where their vehicles are at any time, and communicate updated routes at any time.

Of course, many real-life routing problems contain dynamic and stochastic elements.

In this dissertation, we study two particular distribution problems, both motivated by real-life settings. These problems share many of the common aspects of stochastic and dynamic routing problems, but have also some specific characteristics. In the remainder of the introduction, we introduce the two problems studied and summarize our contributions, and provide a review of the relevant literature on stochastic and dynamic vehicle routing problems.

1.3 A Vehicle Routing Problem with Stochastic Customers and Stochastic Demands

A distributor of alcoholic beverages operating in North Georgia, including the Atlanta metropolitan area, serves a geographically dispersed set of customers (restaurants, convenience stores, gas stations, grocery stores, etc.) using a private fleet of vehicles. Customers order fairly regularly, but not every day; the probability of placing an order on a particular day is known for every day of the week. When a customer does place an order, the order will be delivered the following day. A customer's order quantity also varies from order to order; the order quantity distribution is known for each customer. Given a set of orders for the following day, the distributor constructs delivery routes, picks the items from warehouse storage and packs the trucks overnight for early morning dispatch. The quantity to be delivered on a route cannot exceed the vehicle capacity, the route duration has to conform to driver work rules, and a customer can only be visited within its delivery window. The objective is to minimize the average daily routing cost over some planning horizon. So far, the characteristics of the routing problem described above are quite common. The natural approach would be to use a vehicle routing software package to construct daily delivery routes, because all customer orders are known before the vehicles depart from distribution center. However, the distributor prefers to send the same driver to the same customer as much as possible. Therefore, the distributor dispatches daily routes that are derived from a set of fixed routes.

Fixed routes are delivery routes that are used essentially unchanged for a period of time. Fixed routes are commonly used in practice because they offer many advantages. For example, since the same driver is usually assigned to the same fixed route, drivers familiarize themselves with a region of the delivery area, which results in time savings, especially in big metropolitan areas. Perhaps more importantly, the use of fixed routes can improve customer service as the same driver visits the same customers repeatedly (a more detailed discussion of the benefits of using fixed routes as well as its disadvantages can be found in Chapter 2).

The distributor of alcoholic beverages wants to use fixed routes because its drivers have responsibilities beyond simply delivering the beverages, e.g., they monitor the inventory and they place promotional items. Unfortunately, the order patterns of the customers make it difficult to plan and costly to use traditional fixed routes. First, there is significant variability in the set of customers that order on a particular day of the week and the size of the orders they place. Second, the presence of delivery windows complicate the planning process. In fact, we are not aware of any literature on vehicle routing problems with stochastic customers that considers delivery windows.

In Chapter 2, we propose an innovative, flexible, and practical fixed routes system that can be deployed in settings with medium to high variability and where customers have delivery windows that need to be respected. It can be viewed as a solution to the Vehicle Routing Problem with Stochastic Customers and Demands and with Time Windows (VRPSCD-TW) and our study constitutes the first attempt to address this problem to the best of our knowledge. We introduce a new recovery strategy that relies on limited vehicle sharing, in which customers are assigned to two planned routes, a primary route and a secondary (backup) route. We introduce Monte-Carlo sampling-based techniques to handle the delivery windows at customers during the construction of primary and backup routes, which is necessary to deal with stochastic customers. We develop algorithms that construct primary and secondary routes and present a computational study demonstrating the efficacy of the suggested approaches on real-world data. Finally, we discuss new and promising ideas for more efficient ways to check time feasibility in construction heuristics based on insertion.

In Chapter 3, we study the fixed routes system introduced in Chapter 2 in an abstract and academic setting to gain a better understanding of its core properties. We assume all customer locations are in an interval with the distribution center located at one end point of the interval. By restricting customer locations to be on an interval, the routing aspect becomes trivial and we can concentrate on the vehicle sharing aspect. We focus on gaining insights in the performance of the fixed routes system with limited vehicle sharing compared to the performance of the traditional fixed routes system. For the traditional setting, we characterize the optimal fixed routes system, in terms of expected distribution costs, for instances in which customers with unit demands have order probabilities that are monotone non-decreasing in the distance from the distribution center. For instances in which customers have arbitrary order probabilities or arbitrary order quantities, we provide examples of counter-intuitive fixed routes systems that perform better than intuitive fixed routes systems. For the setting with limited vehicle sharing, we obtain some interesting complexity results. For example, the feasibility of a fixed routes system, i.e., a set of primary and backup routes, can be established in polynomial time. That is, we can establish in polynomial time whether the customers can be served feasibly for every possible demand realization. On the other hand, we show that even if a fixed routes system is feasible, finding a minimum cost distribution strategy for a given demand realization is, in general, NP-hard. However, if the fixed routes system satisfies certain properties, finding a minimum cost distribution strategy for a given demand realization can be done in polynomial time. We also compare, both theoretically and empirically, the performance of certain natural fixed routes systems.

1.4 A Stochastic and Dynamic Vehicle Routing Problem

Consider a company that provides certain services to its customers. For example, a company that performs maintenance and repair services of computer equipment for other companies. Typically, such services are provided under a *Service Level Agreement* (SLA) that specifies a service guarantee, for example, a requested repair will be performed within four hours of the time the request is received. Since requests for repairs arrive over time, the repairmen are dispatched dynamically. Furthermore, the service company may have some knowledge concerning the likelihood of repairs being required at the different locations. Therefore, the dispatching of repairmen falls in the category of a stochastic and dynamic routing problem.

In Chapter 4 we study the core decision of this type of problem in abstract and academic setting. More specifically, we consider an environment in which requests for service arrive at the start of a period and in which the service agreement stipulates that the service has to be performed either in the period in which the request is received or in the subsequent period. Consequently, the dispatcher's decision is whether to serve a request immediately in the period in which it arrives, or whether to postpone it to the next period, in which it will have to be served. Therefore, at the beginning of each period, there are two sets of customers. In the first set, there are the customers that must served in that period because their service was postponed in the previous period. In the second set, are the customers who may be served in this period or in the subsequent period. The dispatcher has to decide which of the request in the second set of customers will be served this period and which will be served in the subsequent period. The dispatcher does not know the exact set of requests that will arrive in the next period, but he knows them in distribution. The objective is to minimize total transportation costs over some planning horizon. To further simplify the problem setting we assume that a single service request arrives each period, so that minimizing the transportation costs is trivial as at most two customers have to be visited in any period. This allows us to concentrate on the dynamic aspect of the problem, rather than on the routing aspect. We study this problem with customer locations on an interval with the depot at an end point, on a circle with the depot in the center, on a disk with the depot at the center (and with a specific distance metric), and on the Euclidean plane with the depot at the origin.

The contributions resulting from our study can be summarized as follows. We show that the optimal policy for the settings in which customer locations are on the interval and on the circle are given by a threshold policy and we compute the optimal policy assuming the customer locations are uniformly distributed. We also propose a simple myopic policy that can be used in all settings, has little computational requirements, and performs surprisingly well. We conduct extensive computational experiments in which we compare the performances of the optimal threshold policy, the simple myopic policy, sampling-based policies, and the online policies that are proposed in the literature for the dynamic version of the problem, i.e., the variant in which there is no information about future customers. The inclusion of online policies allows us to analyze the value of information about future customer orders.

We briefly consider a multi-customer-per-period variant, in which we assume a vehicle with infinite capacity so as to have a traveling salesman problem to determine the transportation cost as opposed to a vehicle routing problem. We introduce Monte-Carlo sampling-based heuristics and test their performance empirically.

1.5 Relevant Literature

Two main approaches have been proposed in the literature to solve SVRPs. The first approach is Chance Constraint Programming (CCP), which was introduced by Charnes and Cooper [19]. CCP is motivated by the observation that it is likely to be too costly to ensure feasibility for all possible realizations of the random variables and that many of these realizations are highly unlikely anyway. Therefore, instead of requiring that a feasibility constraint is satisfied by all possible realizations, the constraint is only required to be satisfied by a pre-specified fraction of all possible realizations, for example by 95% of the realizations. Note that this approach does not consider the costs associated with violating a constraint. The second approach is Stochastic Programming with Recourse (SPR). In SPR, in the first stage, a planned or *a priori* solution is constructed. In the second stage, after the random variables are observed, a recourse or corrective action may be applied to the first stage solution to recover feasibility. The objective is to find a first stage solution and a recourse policy that result in the smallest total costs, i.e., first stage costs plus expected recourse costs. SPR is also known as *a priori* optimization, which was introduced by Bertsimas [12].

Gendreau et al. [32] provides an excellent review of the literature on stochastic vehicle routing problems. It summarizes the various problem settings and the proposed solution approaches. Next, we present a more detailed literature review of the stochastic routing problem variants that are relevant to our work.

The *Vehicle Routing Problem with Stochastic Demands* (VRPSD) is the most studied stochastic routing problem. It arises when customer demand is uncertain and actual demand is revealed only upon arrival at a customer. There are many practical situations that satisfy this condition, for example garbage collection, money collection at ATMs or bank branches, and re-stocking of vending machines. A route failure, a term introduced by Dror and Trudeau [24], is said to occur if total actual demand on a route exceeds the capacity of the vehicle assigned to that route, in which case a corrective action is required. Usually, the corrective action is a trip back to the depot after which the original route is resumed. The objective is thus to minimize the cost of the planned routes plus the expected costs associated with route failures. Tillman [65] was the first to study this problem and modified the savings heuristic of Clarke and Wright [20] to account for stochastic demand. Stewart and Golden [64] presented the first Chance Constraint Programming formulation for VRPSD. They propose a model in which there is a penalty cost per unit of demand in excess of vehicle capacity, but the penalty term does not consider the location of the route failure. Later, Dror and Trudeau [24] proposed a different recourse policy which penalizes route failures more heavily. When a route failure occurs, the vehicle is assumed to go back and forth from depot to each of the remaining customers on the initial route. Therefore, their recourse policy does consider the location of the route failure. Dror and Trudeau develop a heuristic based on the savings heuristic of Clarke and Wright and show that the direction of travel affects the travel cost. Dror et al. [23] describe a variety of recourse policies and models for the VRPSD. They also introduce a new solution framework based on Markov decision processes. At every customer location, the as of yet unvisited customers are resequenced. Due to the large state space, the computational requirements are prohibitive. In fact, Dror et al. [22] state that instances with more than three customers are computationally intractable. The case where vehicle routes are re-optimized at each customer visit was formulated by Bastian and Rinnooy Kan [4] for a single vehicle too. They modify objective functions of the two recourse models proposed by Stewart and Golden [64] and show that when customer demands are identically and independently distributed, the model becomes equivalent to the Time Dependent Traveling Salesman Problem (TDTSP). More recently, Ak and Erera [1] introduce yet another recourse strategy, a paired-vehicle recourse strategy, where vehicles are coordinated in pairs and they suggest a tabu search heuristic for finding high quality solutions. Secomandi [56] proposes another re-optimization strategy based on dynamic programming. He formulates the problem as a stochastic shortest path problem and develops an exact DP model that can solve instances up to 10 customers optimally. Secomandi [57, 59] further employs Neuro-Dynamic Programming (NDP) to stochastic routing problems to eliminate the difficulties associated with large state spaces, where cost expressions are replaced by parametric function approximations. He compares the performance of two NDP algorithms: optimistic approximate policy iteration and roll-out policy on VRPSD and reports that roll-out policy is superior. Savelsbergh and Goetschalkx [55] modify the heuristic by Fisher and Jaikumar [26] to solve the VRPSD and provide insights into the cost benefits of route re-optimization. Haughton [34, 35] further try to quantify the benefits of route re-optimization when customer demand follows a Bernoulli process, i.e., it is either a quantity q_i with probability p_i or zero with probability $1 - p_i$ and customers with zero demand are simply skipped on the planned routes. He discusses various demand stabilization strategies and develops models to estimate the travel distance reductions from these strategies. Intermediate strategies have also been discussed in literature. For example, Yang et al. [68] design a recourse strategy which combines restocking and routing decisions. Instead of waiting for a route failure to occur and then return to depot, the authors determine specific re-stocking points for each customer along each route. They show that in an optimal restocking policy, the vehicle should return back to depot if the remaining capacity drops below a threshold value q_j after serving a customer j and continue with the planned route otherwise. Bertsimas et al. [10], Yang et al. [68], Laporte and Louveaux [47], and Dror et al. [22] also analyze preventive trips to the depot. In addition to the Markov Decision Models cited earlier, which aim to find optimal policies, other exact approaches have been proposed to solve VRPSD as well. Laporte and Louveaux [48] introduce the Integer L-Shaped Method, which is an extension of Benders decomposition [5]. Seguin [60] and Gendreau et al. [31] apply the integer L-Shaped Method for the first time to VRPSD and solve instances up to 70 customers. Hjorring and Holt [36] consider the single-vehicle version of the problem and introduce new optimality cuts to be used in the integer L-Shaped method, which they call "general optimality cuts" since each cut is a bound on the route failure cost for many solutions, in contrast to earlier optimality cuts which impose a bound on the route failure cost of a single solution. Finally, Laporte et al. [46] design an improved integer L-Shaped Method and solve instances up to 100 customers.

In the *Vehicle Routing Problem with Stochastic Customers* (VRPSC) only a subset of customers require an actual delivery. Customer i is assumed to require a delivery with probability p_i and require no delivery with probability $1 - p_i$. Contrary to VRPSD, a customer that does not require a delivery is not visited in VRPSC. The single-vehicle version of this problem, known as the Probabilistic Traveling Salesman Problem (PTSP) was introduced by Jaillet [37]. The goal is to find an *a priori* tour of minimum expected length that starts and ends at the depot visiting each customer once. The *a priori* tour represents the first stage solution, and the recourse policy is tour-skipping, i.e., skipping absent customers. Jaillet [37, 39] examines properties of PTSP and proves asymptotic results for instances in the Euclidean plane. He shows that certain elementary properties of optimal TSP tours in a deterministic setting no longer hold in a stochastic setting; for instance an optimal PTSP tour may cross itself on Euclidean plane. (See Dror et al. [23] for the investigation of these properties in the multi-vehicle context). Jezequel [41], and Rossi and Gavioli [54] adapt the savings heuristic for PTSP and Bertsimas and Howell [11] design a heuristic based on the space-filling curve heuristic for the TSP (Bartholdi and Platzman [3]). The multi-vehicle version (VRPSC) was studied by Bertsimas in his PhD thesis [8]. He derives several bounds, asymptotic results, and other theoretical properties for the case where each demand is equal to 1 with probability p_i and equal to 0 with probability $1 - p_i$. He proposes and performs an asymptotic analysis of a number of greedy heuristics. Jezequel [41] proposes heuristics for both the single- and multi-vehicle versions of the problem. Laporte et al. [49] propose an exact algorithm for PTSP based on the integer L-Shaped method. The authors report that their model is capable of solving instances between 10 and 50 customers and that more random instances, i.e., instances in which customer order probabilities are close to 0, are more difficult to solve than instances with little uncertainty, i.e., where order probabilities are close to 1.

The stochastic routing problem in which customers may or may not place an order and when customers place an order the size of the order is uncertain is known as the *Vehicle Routing Problem with Stochastic Customers and Demands* (VRPSCD). The goal is to determine a set of *a priori* routes of minimal expected total length, where total length consists of the expected length of the *a priori* routes plus the expected length of any recourse actions. Note that the expected length of the *a priori* routes depends on when customer orders are known. If there is no advance information about customer orders, then vehicles will have to visit all the customers in their *a priori* tours; the problem is reduced to VRPSD. However, if there is advance information about customers' orders, then the customers who have not placed an order are simply skipped. VRPSCD is first mentioned by Jezequel [41]. Jaillet [38] and Jaillet and Odoni [40] discuss it and show that the expected total length depends on the direction of travel (even in the symmetric cost case) and that a larger vehicle capacity may yield a larger expected total length. Bertsimas [9] analyzes the situation with advance order information and provides a recursive expression for computing the expected total length of an a prior route, i.e., skipping customers that have not placed an order and returning back to the depot when a route failure occurs. He also proves bounds and asymptotic results. Seguin [60] and Gendreau et al. [31] propose an exact algorithm based on the integer L-Shaped method (Laporte and Louveaux [48]). They conclude that stochastic customers complicate the problem more than stochastic demands. They also report that the problem becomes more difficult to solve optimally as the expected filling rate of vehicles increases (they solve instances of up to 70 customers for a filling rate of 0.3). Their study provides the first comparison of the performance of heuristics against the optimal solution for VRPSCD. Due to difficulty of the problem, heuristic approaches are most popular. Benton and Rossetti [7] develop a multi-stage heuristic, which re-optimizes the *a priori* solution after a customer demand becomes known. Gendreau et al. [33] propose a tabu search heuristic, which is the first time tabu search has been applied to a stochastic routing problem. They assume advance information about which customers place orders, but no advance information about order sizes. Therefore, route failures may occur and are handled using return trips back to depot. Since tabu search requires the evaluation of many solutions within a neighborhood in a short time, they develop an approximation method to evaluate potential moves. They report that the heuristic produces the optimal solution for 89.45% of the instances (with 6 to 46 customers), with an average deviation from optimality of only 0.38%, and that the optimality gap was smaller than 5% in 97.8% of all instances.

In the *Vehicle Routing Problem with Stochastic Travel Times* (VRPSTT) customer demands are known with certainty, but the travel time between two locations is uncertain. The travel time between two locations not only depends on the distance between them, but also on the vehicle speed, which is affected by traffic conditions, weather conditions, and road conditions. In practice, travel times are rarely deterministic, whereas almost all the vehicle routing literature assumes that travel times are directly proportional to the distance traveled and assumes a constant velocity. Leipala [51] analyzes the expected length of an *a priori* route assuming arc lengths are random. Kao [42], Sniedovich [62], and Carraway et al. [18] consider PTSP with an objective of maximizing the probability of completing an *a priori* tour by a given deadline when the arcs have independent and normally distributed travel times. Laporte et al. [45] introduce the SVRP with stochastic travel and service times. In their study, it is assumed that each customer has to be served and each vehicle has a target time by which its route should be completed. They propose two models: a chance constraint model which ensures that each route completes its service by the predetermined time with some minimum probability, and a stochastic program with recourse where there is a penalty proportional to the expected length of the delay. Instances up to 20 customers were solved optimally for the stochastic program with simple recourse using a branch-and-cut approach. Lambert et al. [44] studies this problem in the context of money collection from bank branches and penalizes late arrivals. They propose an adaptation of the savings algorithm and present results for instances with 28 and 44 customers and with two different travel times (all long or all short). Kenyon and Morton [43] propose two models with different objectives. The first model minimizes the expected completion time, where completion time is the time the last vehicle returns to depot. The second model maximizes the probability that all service is completed before a pre-determined time. They report that a 28-customer instance with two vehicles and continuous random parameters can be solved optimally by a branch-and-cut algorithm. Finally, Verweij et al. [67] apply the sample average approximation method to solve a single vehicle version of this problem, where the expected value of the objective function is approximated by a sample average estimate derived from a sample of random realizations.

An important and practically relevant variant of VRP is the Vehicle Routing Problem with Time Windows (VRPTW), in which a customer's service has to start within a given time window. Although VRPTW has been studied extensively (see Cordeau et al. [21] for a survey), the same is not true for stochastic variants of the VRPTW. In fact, there are only a few recent papers that address time in stochastic routing problems. Campbell and Thomas [17] introduce the PTSP with Deadlines (PTSPD), where customers have to be served before a pre-specified deadline. Three models are presented. In the first model, all realized customers are visited, but a penalty is incurred when violating a deadline. In the second model, any realized customer whose deadline would be violated is skipped, but a (different) penalty is incurred. The third model is a chance-constraint model in which all realized customers are visited with a probabilistic constraint on the violation of a customer's deadline. The authors compare TSP with deadlines (TSPD) and PTSPD through a series of computational experiments. They present a number of interesting observations. For example, when there is no feasible solution for TSPD with respect to deadlines when all customers require a visit, then modeling this problem stochastically greatly impacts the solutions and the impact is greater when customers have low order probabilities. In contrast, if a feasible solution exists and customers have high order probabilities, then modeling stochastic customers is not that critical. Even more interestingly, they report that stochastic modeling is desirable when both high and low probability customers exist and they attribute this to the fact that stochastic modeling gives greater importance to customers with high probabilities. Morales [52] reports results on the VRPSD with time window constraints. He shows that the worst-case demand realization for a tour can be identified by solving a longest path problem on an acyclic network and proposes a tabu search heuristic.

In the *Dynamic Vehicle Routing Problem* (DVRP), customer orders are not known prior to the determination of the vehicle routes but are revealed over time. Vehicle routes may be updated to accommodate newly arrived customer orders. In addition to the traditional objective of minimizing total travel distance, other objectives are also considered, e.g., maximizing the number of customers served or minimizing the average waiting time. Psaraftis

[53] provides a comprehensive survey of early work on DVRP and discuss a number of applications. The PhD thesis of Larsen [50] also contains a thorough review of DVRP. The specific dynamic routing problem studied in this dissertation is introduced in Angelelli et al. [2]. They study the competitive ratios of various simple dispatching policies. The performance of a dynamic routing algorithm is sometimes measured by its competitive ratio, a notion introduced by Sleator and Tarjan [61] for the analysis of online algorithms.

Stochastic and dynamic routing problems include both stochastic and dynamic characteristics. The seminal paper by Bertsimas and Van Ryzin [14] introduces the *Dynamic Traveling Repairman Problem* (DTRP). Customer orders arrive over time according to a Poisson process and require an independent and identically distributed service time. The objective is to minimize the average waiting time for service. Examples are provided of applications where minimizing waiting time is more important than minimizing travel costs. They propose several heuristics, e.g., as First Come First Serve (FCFS), Nearest Neighbor, Partitioning, and Traveling Salesman Policy where customer demands are collected into sets and then served using an optimal TSP tour, and they analyze their performance for different arrival rates. Bertsimas and Simchi-Levi [13] presents a comprehensive survey of DTRP. Recently, Secomandi [58] proposes a rollout policy which can be used in a dynamic and stochastic environment. The policy revises a previously computed static solution every time new information becomes available. Finally, Bent and Van Hentenryck [6] studies a dynamic VRPTW with stochastic customers in which the goal is to maximize the number of served customers. The authors present a Multiple Scenario Approach (MSA) which is an extension of the Multiple Plan Approach (MPA) developed by Gendreau et al. [30] for dynamic VRPTW and show that MSA provides large improvements over approaches that do not use stochastic information.

CHAPTER II

FIXED ROUTING SYSTEM FOR STOCHASTIC ENVIRONMENTS

2.1 Introduction

In this chapter, we propose a practical and flexible fixed routing system that preserves many of the benefits of traditional fixed routes but can be deployed in settings with medium to high variability and delivery time window constraints. We introduce a new recourse strategy, in which customers are assigned to *two* planned routes, a primary and a backup, and recourse decisions can move customers to backup routes to regain feasibility or improve costs, and the use of sampling-based techniques to handle the presence of delivery time windows during the construction of primary and backup routes. We also present a computational study based on real-life data to demonstrate the efficacy of the proposed fixed routing system and the route construction techniques.

We specifically consider a vehicle routing system where a vehicle fleet operates daily delivery routes from a depot. Each day, the set of customers to be visited is a subset of the entire customer base. Furthermore, the quantity delivered to a specific customer may vary for each day a delivery is made. When probabilistic information is available describing delivery request likelihoods and demand quantities, the associated planning problem falls into the category of *stochastic vehicle routing problems.*

Fixed routes, i.e., daily delivery routes that are used essentially unchanged for a period of time, are commonly used in practice. In a typical fixed routes solution, each customer in the customer base is placed on an ordered route for some vehicle. Operational routes for each vehicle are determined by visiting customers in the order prescribed by the fixed routes, skipping customers that do not need a visit that day. Additional adjustments (or recourse decisions) may be made to ensure that the operational routes are feasible and practical.

Fixed routes offer numerous advantages. Using fixed routes may require lower total costs than daily optimized routes for several reasons. First, fixed routes enable simplified and streamlined loading operations at the depot. Second, they allow drivers to familiarize themselves with a region of the delivery territory which often results in time savings. Finally, they simplify the daily planning process and eliminate the need for route optimization software and (skilled) personnel to effectively use that software. Another equally if not more important advantage of fixed routes is that they can improve customer service. Customers may typically be visited at or around the same time each day, which allows each customer to adjust its processes to accommodate the delivery. Since fixed routes enable the same driver to visit the same customers repeatedly, drivers establish long-term relationships with customers. These relationships can be useful, for example, when unforeseen circumstances cause a delivery to be late and the driver must call ahead to notify the customer. Additionally, drivers are often responsible for the actual stocking of the products to shelves, and for monitoring the inventory. In such situations, the store owner needs to trust the driver to do a good job, and such trust is established over time.

Of course, there are also disadvantages to using fixed routes. Routing costs may be higher than when using daily optimized delivery routes. Furthermore, the rigidity of the routes may lead to under-utilized vehicles and unbalanced driver workloads; this may be especially true if recourse options are limited, and in settings with relatively high day-to-day customer demand variability.

The study in this chapter is motivated by our collaboration with a beer, wine, and spirits distributor in the Atlanta area. The distributor would like to employ fixed routes for its distribution operations, but important system characteristics make it difficult to plan and costly to use traditional fixed routes in this case. First, the set of customers requesting delivery on any given day is highly variable, and the demands placed by these customers when they do request delivery vary significantly. Second, all customers have visit time windows; these windows are often narrow, and some customers have two visit windows each day. Third, each customer usually places at most one order per week, but sometimes more orders are placed and the order weekday sometimes varies. Rigid fixed routes planned to be repeated daily with acceptably low failure probabilities in this setting would require far too large a vehicle fleet.

With this motivation, we propose a more practical and flexible fixed routing system that preserves many of the benefits of traditional fixed routes but can be deployed in settings with medium to high day-to-day customer variability and difficult time constraints; most real-world routing systems have these features. The three key ideas in our approach are: (1) the customer base placing orders on each weekday is partitioned into two subsets, *regular customers* who place orders frequently on that day and thus are included on planned routes and *irregular customers* who are served infrequently and are therefore only added to operational routes dynamically as necessary; (2) regular customers are assigned to *two* planned routes for each weekday, a primary and a backup, and recourse decisions can move customers to backup routes to regain feasibility or improve costs; and (3) the order of customer visits suggested by the planned routes can be changed by recourse to improve costs.

The paper will develop optimization technology that (1) constructs a set of planned routes, primary and backup, which leads to cost-effective daily operational routes, and (2) constructs a low-cost set of operational routes for a given customer demand realization given a set of planned routes. The problem of constructing a set of planned routes is a complex variant in the class of stochastic vehicle routing problems. An important contribution of this research will be the development of techniques to incorporate delivery time windows in the construction of fixed routes in stochastic routing. An equally important contribution will be the introduction and analysis of a new recourse strategy based on the use of backup routes. Neither of these features has been studied to our knowledge in the stochastic routing literature.

The remainder is organized as follows. In Section 2.2, we formally define the problem that we will study. In Sections 2.3 and 2.4, we discuss our methodology for constructing a set of primary and backup planned routes respectively. In Section 2.5, we illustrate how daily operational routes are constructed using planned routes. In Section 2.6, we present an extensive computational study demonstrating the viability and effectiveness of our proposed methodology. Finally, in Section 2.7, we introduce future research directions.

2.2 Fixed Routes with Backup Vehicles: Problem Definition

Consider a distribution problem from a single distribution center. A distributor uses a fixed fleet of homogeneous vehicles to serve customers each day. Each day, each customer may or may not place a delivery order, and the distributor delivers the order the following day. A customer's order quantity may vary each day an order is placed. Given the full set of orders for the following day, the distributor constructs operational delivery routes, then picks items from warehouse storage and packs trucks overnight for early morning dispatch. Each individual operational route must be designed such that vehicle capacity is not exceeded, total route duration conforms to driver work rules, and each customer is visited within its allowable delivery time window; a route satisfying all three conditions is considered *feasible*. The objective is to minimize the average total daily operational routing costs over a long horizon.

The distributor would like to dispatch operational routes that are derived from fixed routes, primarily to capture the customer service benefits of having the same driver visit the same customers repeatedly. In the setting that motivates this research, the driver is responsible for moving inventory from the truck to the stores and often is also responsible for placing product on shelves and setting up promotional displays. Since most customers order at most once per week and tend to (but do not always) order on the same weekday when they do place an order, the distributor would like to construct a different set of fixed routes for each weekday such that a customer will see the same driver each time they place an order on the same weekday.

While the distributor desires the benefits of fixed routes, the nature of its business and customer base would necessitate a very large vehicle fleet under a traditional fixed routes solution. A significant fraction of the customers served on a given weekday place an order less than 10 percent of the time on that day throughout the year. Demanded quantities also exhibit significant variation for each customer given an order. After discussions with the company, it was decided therefore that a reasonable system would consider two categories of customers for each weekday: (1) customers that may be served by any driver, and (2) customers that must be served by no more than two different drivers on that weekday.

Customers in category (1) will be those with very low likelihood of placing an order on the specific weekday, and will be added as needed to any operational route. Each customer in category (2) will be included on one or two planned routes for the weekday; the first route will be denoted the *primary* and the second route the *backup*.

Given planned routes and a realization of daily customer demand requests, the recourse (operational) problem will be to determine low-cost, feasible operational routes serving all customers, and further such that each category (2) customer is served either by its primary or backup vehicle. An additional desirable feature is to serve as many category (2) customers as possible with their primary routes. Each resultant operational route, therefore, may include some customers for which this route is primary, some customers for which this route is secondary, and some category (1) customers. Note that we do not require preservation of the planned *order* of customer visits during recourse; this notion is not well-defined given that the operational routes are blends of two planned routes. Furthermore, note that this recourse problem may not have a feasible solution for all possible demand realizations; our goal will be to design planned routes and a recourse problem solution strategy that leads to feasible solutions for nearly all realizations.

We now introduce notation to describe the planning and operational routing problems to be considered in this paper. Since we will assume that independent planned routes are to be developed for each weekday, we will describe the problems to be solved for a single such day. Let $V = \{1, ..., n\}$ be the customer base that may request delivery on a given weekday, and suppose it is partitioned into subsets V_1 and V_2 representing the category (1) and category (2) customers, respectively. Let customer 0 refer to the vehicle depot/distribution center which houses the fleet of m delivery vehicles. Let p_i be the known probability that customer i requests a delivery on the given weekday, and let q_i be a discrete random variable with known probability mass function representing the delivery quantity for customer i given a delivery request. Let Q be the maximum quantity that can be delivered by each vehicle, and assume that $q_i \leq Q$. Let $\hat{q}_i$ be the random variable representing the delivery quantity for customer i (which may be zero if no order placed). Given these parameters, one can derive $\mu_i = p_i E[q_i]$ and $\sigma_i^2 = p_i E[q_i^2] - p_i^2 E[q_i]^2$, the expectation and variance of $\hat{q}_i$.

Each customer i has one or two delivery windows, $[e_i^1, \ell_i^1]$ and $[e_i^2, \ell_i^2]$ ($e_i^2 > \ell_i^1$), where service can begin no earlier than e_i^j and no later than ℓ_i^j; a vehicle arriving at i prior to e_i^1 must wait until e_i^1 to begin service, and one arriving after ℓ_i^1 and prior to e_i^2 must wait until e_i^2 to begin service. Travel times t_{ij} and travel distances d_{ij} are known between each pair of locations $i, j \in V \cup \{0\}$. For convenience, we assume that travel cost from i to j is also equal to d_{ij}, but other cost functions are possible.

Given this setting, we consider two optimization problems. One is the operational problem: given customer demand realization ω and a set of planned routes, determine a set of no greater than m feasible operational routes serving all customers with minimum total travel cost $z(\omega)$. The other problem is a two-stage stochastic optimization problem: determine planned routes, primary and backup, that minimize the expected cost of the operational problem, $E_\omega[z(\omega)]$. In this research, we will develop heuristic solution approaches for both of these hard optimization problems.

2.3 Constructing Planned Primary Routes

We consider the problem of constructing planned routes, both primary and backup, for customer set V_2. It is true, of course, that the feasibility and quality of these routes strongly depend on how they are used by the recourse policy that determines daily operational routes. Since our recourse policy is the result of solving a complex recourse problem, it is not computationally tractable to assess planned route feasibility and quality exactly during a construction process.

We first present our approach for building planned primary routes. We begin with single-customer routes serving m seed customers and then insert remaining customers into routes one by one (see Bräysy and Gendreau [16]). We also use local search during our construction process periodically to improve the route set. As it can be seen, the main ingredients of our approach are similar to those found in standard insertion-based construction procedures for vehicle routing problems with time windows (VRPTW). However, at a detailed level there are substantial differences. The main difference, and a great challenge, is defining and verifying route feasibility. The fact that customers are present on any given day only with a certain probability and that customers have delivery windows that have to be respected significantly complicates assessing and even defining the feasibility of a set of delivery routes, which we discuss next. A complete description of the heuristic is given in Algorithm 1.

Algorithm 1 Heuristic for Constructing Planned Primary Routes

```
Build m initial single-customer routes to m seed customers
Attempt to insert difficult-to-serve customers one-by-one considering all possible insertion
locations
Run improvement procedure
while Insertion failures have not occurred in K consecutive iterations, where K is twice
the number of remaining uninserted customers do
  Attempt to insert a remaining uninserted customer, selected at random, using insertion
  locations defined by its neighbor list
end while
Run improvement procedure
if Some customers are not inserted then
  Increase angular insertion limit
  Attempt to insert customers one-by-one considering all insertion locations
  while All customers not inserted and time feasibility parameter not at minimum do
    Decrease time feasibility parameter
    Attempt to insert customers one-by-one considering all insertion locations
  end while
end if
Run vehicle reduction procedure
```

2.3.1 Feasibility assessment of primary routes

For simplicity, we assess the feasibility of primary routes as if they were to be operated as traditional fixed routes, applying only the skipping recourse strategy for customers not requiring a visit. Since this assessment ignores secondary vehicles and uses a simple (and standard) recourse policy, the results here are applicable to more traditional fixed routing problems with time constraints. This is also a pragmatic choice, since if we design primary routes with reasonably high probabilities of stand-alone feasibility, more customers are likely to be served by their primary vehicles.

For a set of operational routes to be feasible on a particular day, neither vehicle capacity nor customer time windows may be violated. We discuss these two components of feasibility separately, beginning with capacity feasibility.

Recall that customer i places an order on any given day with probability p_i, and requests random quantity q_i when he does. The actual order quantity, $\hat{q}_i$, then is a discrete random variable likely in practice to have a probability mass function with a mass of $1 - p_i$ at zero along with masses scattered at larger values around the conditional mean $E[q_i]$. Given a planned route serving customer subset R, the probability that the route is feasible with respect to vehicle capacity is

$$P\left(\sum_{i \in R} \hat{q}_i \leq Q\right). \tag{1}$$

Since it is likely to be difficult to determine an exact cumulative distribution for the random variable $\sum_{i \in R} \hat{q}_i$, we assume instead that the random variables $\hat{q}_i$ are mutually independent, and use a normal approximation under the central limit theorem. Under independence, such an approximation $D(R)$ to the cumulative demand of route R has mean $\mu(R) = \sum_{i \in R} \mu_i$ and variance $\sigma^2(R) = \sum_{i \in R} \sigma_i^2$. A planned primary route then is considered capacity feasible if

$$P(D(R) \leq Q) \geq \alpha, \tag{2}$$

where α is the approximate capacity feasibility probability, for example $\alpha = 0.9$. Since $D(R)$ is normally-distributed, the route is feasible if

$$Q \geq \mu(R) + \Phi^{-1}(\alpha)\sigma(R), \tag{3}$$

where $\Phi^{-1}(x)$ is the inverse of the standard normal distribution. Thus, checking capacity feasibility of a route in this case is almost as simple as for the standard VRPTW. Note that the central limit theorem requires the number of customers in customer set R to be large and the assumption is weak otherwise. Fortunately, if the number of customers assigned to a fixed route is small (which is the case in the early stages of an insertion based algorithm), vehicle capacity is usually not restrictive.

Time feasibility of a planned primary route is more difficult to analyze. Our goal is to compute the likelihood that a given primary route can be operated such that each customer i is served at time a_i, where $a_i \in [e_i^1, l_i^1] \cup [e_i^2, l_i^2]$, and then to constrain this likelihood to be greater than or equal to some acceptable level β. Suppose a vehicle visits customers

$R = \{1, 2, ..., t\}$ in order. If each $p_i < 1$ for $i \in R$, then there are 2^t possible operational route realizations. Let P_ω be the probability of realization ω:

$$P_\omega = \left(\prod_{i \in R(\omega)} p_i \right) \left(\prod_{j \in R \setminus R(\omega)} (1 - p_j) \right),$$

where $R(\omega)$ is the set of customers requesting service in realization ω. Further, let I_ω be 1 if a time window feasible vector $\{a_i\}$ can be found for customer subset $R(\omega)$, and 0 otherwise. Then, the probability that a primary route is time window feasible is given by $\sum_\omega P_\omega I_\omega$. Clearly, checking for the existence of a feasible $\{a_i\}$ for all 2^t realizations will be impractical for all but the smallest values of t.

Computational efficiency of feasibility checking is critical for construction and local search heuristics; unless such checks can be performed in a reasonably efficient way, the procedure will require too much time to determine very good solutions. A simple, straightforward way to check time window feasibility approximately is via a Monte Carlo sample. We use this approach, and generate N sample realizations of the customers in R requesting service using the probabilities p_i. Route R is considered time feasible if a feasible vector $\{a_i(\omega)\}$ exists for at least βN customer subset realizations $R(\omega)$.

2.3.2 Selecting seed customers

Various methods for selecting seed customers in vehicle routing problems appear in the literature. Bramel and Simchi-Levi [15] identify m seeds by solving a capacitated location problem, in which the sum of the distances of the customers to their closest seed is minimized subject to a limit on the total demand associated with each seed. The early sequential insertion heuristics in Solomon [63] select as seeds customers that are furthest from the depot or those with smallest l_i^1. The idea is to select as seeds customers that will be difficult to insert feasibly into a partially-constructed route. Unlike Solomon's heuristics, we select m customers as seeds and then insert remaining customers into any route, rather than building one route at a time; for this reason, it is not critical to consider l_i^1 when selecting seeds. Importantly, however, since seeds have some influence on the shape of the final solutions, it may not be wise to use infrequently-visited customers. We take a pragmatic approach, partly inspired by the observations above, and choose a representative historical delivery day with not less than m routes. We then select as seeds the farthest customer with $p_i \geq 0.25$ from each of the m delivery routes with the longest durations.

2.3.3 Insertion

Starting with a set of single-customer seed routes, we insert remaining customers sequentially into the partial routes. Like any insertion heuristic for routing, two decisions guide the process: (1) which customer to insert next, and (2) where to insert the chosen customer. A common greedy approach, sometimes labeled *cheapest insertion*, is to compute the least-cost insertion for each un-inserted customer, and then execute the customer insertion with the smallest cost. Due to the computational requirements of our sample-based feasibility assessment, this classic approach is computationally prohibitive for reasonable sample sizes.

To speed up computation, we made two modifications. First, we insert customers into the partial solution in a predefined order, such that in each iteration only the least-cost insertion for a single customer must be determined. Second, in the latter stage of the procedure, when the partial solution includes many customers (and thus many potential insertion positions), we only consider insertion positions before and after members of a *neighbor list* for the customer to be inserted. Thus, the procedure consists of two phases. In the first phase, difficult-to-serve customers are inserted considering all insertion positions. In the second phase, remaining customers are inserted using neighbor lists. A customer is considered difficult when its latest delivery time is noon or earlier, when the available time for a delivery is less than 4 hours (which may be split over two delivery windows), or when it is further than 40 miles from the distribution center. Customers in the first phase are processed in order of non-increasing distance from the distribution center, while customers in the second phase are processed in random order. We define the neighbor list for customer i as follows: (a) all customers no further than 10 miles from customer i, or (b) if customers of type (a) number fewer than 50, the 50 nearest customers to i.

Initial experimentation revealed that, especially in the first phase, some customers were

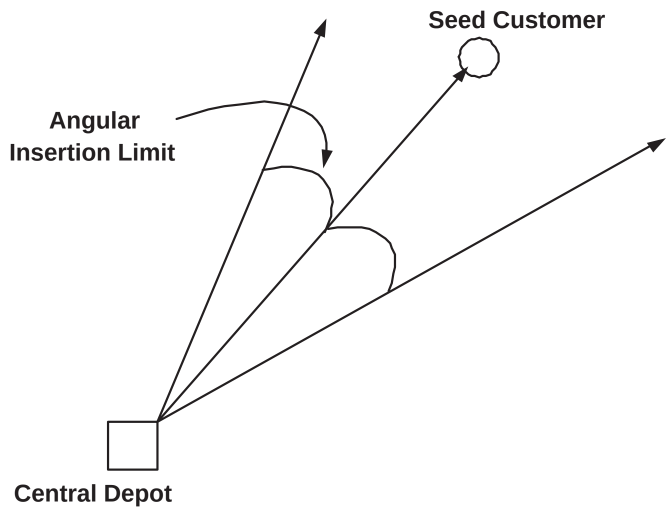

Figure 1: The angular insertion limit restriction

inserted in undesirable positions in the partial solution; these undesirable insertions were primarily due to the pre-defined customer insertion order. Such insertions can be avoided by postponing consideration of these customers such that the local search has improved the partial solution. Postponing is accomplished by allowing only insertion into a route where the absolute difference in polar angle between the customer and the seed customer is at most γ degrees, where angles are measured with the distribution center as the origin. We refer to this restriction as the *angular insertion limit*; see Figure 1. We analyze the impact of different angular insertion limits on the performance of proposed approaches in Section 2.6.1.

We assess a candidate customer insertion for feasibility using the ideas outlined in Section 2.3.1. Importantly, when evaluating insertion positions for customer i in primary route R, we generate a conditional sample such that i is present in each realization $R(\omega)$. Note that if we did not use such an approach and i has relatively small p_i, then that customer may not appear in most of the realizations. A conditional sample seems appropriate since the question we are attempting to answer via insertion is: given that customer i places an order, on what route should it be served? To choose among feasible insertion options, we measure *route quality* by a weighted combination of expected duration and expected travel time; we use a weight of 1 for route duration and a weight of 4 for travel time in our computations. Route duration differs from travel time since waiting may occur at a customer. Since exact computations are computationally prohibitive, we again use a conditional sample to compute a sample average duration and average travel time for a route given an insertion. Note that these averages are computed using each of the N realizations in the sample, including those realizations which are not time feasible.

The average duration of a route can increase significantly in the early iterations of an insertion heuristic, especially if a customer is inserted at a position that creates a significant amount of waiting time. However, not all such insertions are bad choices, since it may be possible to later insert customers in between to reduce or eliminate the waiting. In fact, such insertions might be preferable to those that require longer travel time and less waiting time, since the travel time will always be unavoidable. Therefore, for a primary route R for a given customer realization ω, we set its duration to be $max\{T, \Delta_R(\omega)\}$, where $\Delta_R(\omega)$ is its actual duration and T is a minimum allowed duration; we use $T = 6$ hours in our computational work.

2.3.4 Improvement procedure

We use a simple scheme to improve a partial solution, which we denote k-REINSERT. During each improvement iteration, we randomly select a fixed route R with more than k customers, we remove k consecutive customers randomly from R, and we reinsert them one-by-one using the methods in Section 2.3.3; note that reinsertions into R are allowed. Due to our sample-based approach, it is possible that we do not find a feasible reinsertion for all k customers; in that case, the original route R is restored. For efficiency reasons, we only consider "promising" sequences, where a promising sequence is one where the ejection of the customers in the sequence results in an improvement in route quality of at least T'; we use $T' = 50$ minutes in our computational work.

2.3.5 Feasibility relaxation phase

If all customers cannot be feasibly inserted, we relax the time feasibility requirement, by adjusting β, and the angular insertion limit, by adjusting γ; note that in our application, capacity feasibility is rarely constraining, but if it were, it could be relaxed as well by adjusting α. First, we relax γ to $\gamma_L > \gamma$; the default parameters we use in computations are $\gamma = 15$ and $\gamma_L = 30$ degrees. If all customers still cannot be feasibly inserted, we relax time feasibility parameter β by decreasing it from its initial value in steps of size 0.05 until a minimum level $\beta_{\min} = 0.5$ is reached. If we still have not inserted all customers feasibly at this point, we give up; additional vehicles or different seed customers are probably necessary.

Gradual relaxation of feasibility requirements is an important algorithmic choice. If initial feasibility requirements are set too high, then finding a set of fixed routes that satisfies those requirements may be difficult. On the other hand, if initial feasibility requirements are set too low, then bad insertions can be accepted in the early stages of the algorithm, which can make inserting remaining customers difficult. Therefore, feasibility requirements are relaxed gradually if the algorithm faces difficulty inserting customers into fixed routes.

2.3.6 Vehicle reduction procedure

To this point in the heuristic, we have fixed the number of vehicles (and thus the number of constructed primary routes) to be m. As a final phase, we attempt to eliminate under-utilized vehicles. Starting with the primary route with the minimum average duration, we eject all of its customers and attempt to reinsert them into other routes, restricting ourselves to routes that satisfy the original feasibility requirements. If successful, the route is eliminated and we proceed to the remaining primary route with minimum average duration. If unsuccessful, the original route is restored and the phase ends.

2.4 Constructing Planned Secondary Routes

Next, we consider an approach for constructing planned secondary routes for the customer set V_2. Recall from Section 2.3 that primary routes are planned using feasibility targets α and β which approximate the feasibility likelihood for each primary route given that it is operated using the skipping recourse strategy for customers not requiring a visit. In practice, the goal is to achieve operational feasibility for nearly all realizations. In this section, we describe an approach for constructing planned secondary routes for achieving this goal. In our approach, a secondary "route" is not a route per se, since the set of customers assigned to a common secondary vehicle is not ordered.

We again develop a sample-based procedure, and use a consensus approach to determine best secondary vehicle assignments. For some sample size N_2, we generate N_2 realizations of customers requesting delivery from the complete customer set V using probabilities p_i. For each customer i in each realized set $V(\omega)$, we generate a demand realization $q_i(\omega)$ using the probability mass function for q_i assuming that q_i is normally distributed with known mean and variance and by rounding to nearest non-negative integer. Note that there is a positive probability that a given $i \in V_2$ may not appear in any of the realizations, but for reasonable values of p_i and N_2 this probability $((1-p_i)^{N_2})$is very small.

We now describe the secondary route generation approach. For each realization ω, we first apply the skipping recourse strategy to the primary routes to generate tentative operational routes serving the customers from the set $V_2(\omega)$ that requested delivery. If all tentative routes are feasible, no information is generated and we move on to the next realization. If one or more routes are infeasible, we attempt to first recover feasibility by repeatedly executing reinsert moves. Each reinsert move selects a customer at random on an infeasible route to eject and reinsert into another route. To select among feasible insertion positions, we select the candidate that minimizes the change in route quality. (All possible insertion positions are evaluated, i.e., no neighbor lists are used.)

The recovery process terminates when all operational routes are feasible, or if $2|R|$ reinsert moves have failed for some infeasible route R with $|R|$ customers. If the operational routes are not all feasible at termination, we again gain no information and we move on to the next realization. Otherwise, we next improve the operational routes using a simple local search. For some number of iterations (300 in our computational experiments), we eject a customer at random from its route and attempt to reinsert it such that the total quality of the affected routes is improved. Note that this procedure may empty some operational routes completely. Finally, we attempt to insert customers in the realization from set $V_1(\omega)$; recall that these customers have very low probabilities of requesting service, and are close to the depot. We begin with the furthest such customer from the depot, and insert them sequentially considering all feasible insertion positions differentiated by marginal route quality.

If the realization results in a feasible delivery schedule for all arriving customers, we record the vehicle number that serves each customer $i \in V_2(\omega)$. After performing the steps on each realization in the sample, we have for each customer i a list of vehicles that were used to serve the customer, and a count of the number of realizations in which vehicle k was used. The vehicle with the highest count, not including its primary vehicle, is chosen as the secondary vehicle for customer i (ties are broken arbitrarily). If the primary routes have been constructed well, the vehicle with the highest count is likely to be the primary vehicle. Therefore, this procedure can also be used to validate the primary routes. It is possible that a customer is always visited by its primary vehicle. In that case, no secondary assignment is made.

2.5 Constructing Daily Delivery Routes

Finally, we propose an approach for solving the recourse problem, i.e., the problem of determining daily operational routes given the primary and secondary route assignments. Note first that determining an optimal solution to this problem where the objective is to minimize total travel time or route duration is clearly NP-hard by reduction from the deterministic traveling salesman problem.

Our goal is to develop a very fast heuristic that takes advantage of the fact that the customer visit sequences given by the primary routes should represent a very good starting point. Thus, this heuristic is similar to the one used in Section 2.4 to determine secondary routes.

We again first apply the skipping recourse strategy to the primary routes to generate tentative operational routes serving the arriving customers from the set V_2. If the resulting solution is infeasible, we attempt to recover feasibility by repeatedly executing reinsert moves for customers on infeasible routes, with the additional restriction that an ejected customer can only be reinserted into the route for its secondary vehicle. Assuming that this simple reinsertion always restores feasibility, the schedule is next improved via local search. For a fixed number of iterations, we randomly select a customer, remove it from its current route (primary or secondary), and attempt to reinsert it in its alternate route (secondary or primary) such that route quality is improved. Finally, each customer is removed and reinserted into its current route in an attempt to improve the final route sequence. (All possible insertion positions are evaluated, i.e., no neighbor lists are used.)

Next, we need to insert arriving customers from V_1. This is performed by sequential insertion in order of non-increasing distance from the distribution center to the insertion position that minimizes marginal increase in route quality.

Finally, a vehicle reduction procedure is employed similar to the one used as the final phase of the primary route construction. Vehicles are considered one-by-one, in non-decreasing order of their return times to the distribution center. All customers are removed and reinserted sequentially in non-increasing order of distance to the distribution center; note that primary and secondary vehicle assignments must be respected for customers in V_2. If unsuccessful, the original vehicle route is restored and the procedure stops.

2.6 Computational Study

We now present the results of a computational study that was conducted using historical demand data obtained from a distributor of alcoholic beverages to validate and analyze the efficacy of the proposed solution methods. Our computational study is divided into two parts. In the first part, we analyze the impact of key heuristic parameter values on the performance of suggested algorithms. Specifically, we focus on the primary route time feasibility requirement (β), sample size (N), and the angular insertion limit (γ). Because vehicle capacity is rarely limiting in this environment, the capacity feasibility requirement parameter (α) is not varied. In the second part, we compare the performance of the solutions resulting from our proposed methods to the solutions currently deployed by the distributor to see the improvements achieved. All algorithms were implemented in C and executed on an Intel Xeon 2.4 GHz processor with 2 GB of memory.

We first summarize the basic characteristics of the problem environment. There are 4,356 customers in the service region, which includes a major US metropolitan area. The single distribution center is centrally-located in the region, and while 95% of customers are concentrated within a 70 mile radius of the depot, some are scattered to as far as 137 miles

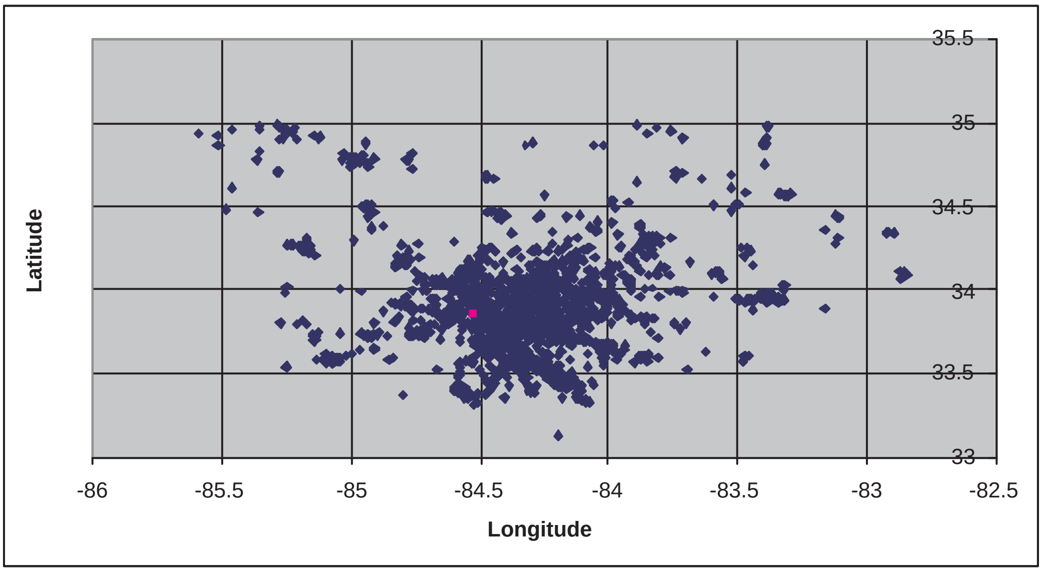

Figure 2: Customer Locations

(Figure 2).

For each customer and for each day of the week, we use historical ordering information to determine an estimated probability that a visit will be requested on that day. Table 1 summarizes this information; note that customers who have never historically ordered service for a specific day are assumed to have zero probability of a future order on that day. Based on this information, we decided, in conjunction with our industry partner, that the regular customer set (V_2) for a given weekday should include customers with an order probability greater than 0.1; since customers far from the depot are also difficult to serve dynamically, we also include in V_2 customers located outside a 40 mile radius. From the table, it is also clear that the number of customers to be served on Mondays is likely to be substantially smaller than on other weekdays, which is a demand characteristic common to alcoholic beverage distributors.

Historical order quantity information is used to derive estimates of the mean and variance of each customer's order size for a given weekday, given that an order is placed; demand is standardized to units of *cases* using appropriate conversion factors. On Monday, the average mean demand of customers in V_2 is 15.08 cases, with a minimum mean demand of

1 case and a maximum mean demand of 277 cases. The numbers are similar for the other days of the week.

Table 1: Customer Order Likelihood Summary: for each week day, the table provides the number of customers that fall within the specified probability range.

Probability Range	M	Tu	W	Th	F
$p = 0$	3504	2282	2170	1839	1858
$0 < p < 0.1$	522	621	820	870	925
$p >= 0.1$	**330**	**1453**	**1366**	**1647**	**1573**
$p >= 0.2$	209	1130	988	1198	1130
$p >= 0.3$	173	936	786	966	909
$p >= 0.4$	159	823	639	801	767
$p >= 0.5$	141	735	528	698	659
$p >= 0.6$	105	605	397	540	492
$p >= 0.7$	95	536	358	474	439
$p >= 0.8$	78	464	301	406	365
$p >= 0.9$	63	388	247	325	280
$p = 1$	39	268	167	206	179

Each customer has one or two feasible delivery windows that can be used on service days. For the distributor studied, there is great variety of delivery windows among customers; there are 194 different delivery window configurations found in the data. Nine of these configurations, however, are used by 75% of the customers; see Table 2. The most popular configuration allows deliveries during a two-hour morning window and a four-hour afternoon window. Some customer windows are quite tight; 10% allow deliveries for only four hours during the day. The customer data also includes an estimate of the time required to perform service, and the travel times between pairs of customers. Service times range from 20 to 35 minutes, and travel times range from 0 (for co-located customers) to a little over 4 hours for the two customers with the largest distance between them.

A fleet of 45 vehicles is based at the distribution center, each vehicle with a capacity of 700 cases, is available for deliveries. Each vehicle can perform one tour per day, departing from the distribution center as early as 4 am, but returning to the distribution center no later than 8 pm.

Since customer demand is highest historically on Thursdays and lowest on Mondays, we focus the computational experiments presented to these two days. To evaluate the performance of the proposed methodology, we focus on efficiency (run time) and effectiveness

Table 2: Customer Delivery Windows Summary: the top nine delivery window configurations

Early 1	Late 1	Early 2	Late 2	Number of Customers	Percent
9:00 AM	11:00 AM	2:00 PM	6:00 PM	1240	28.47
8:00 AM	4:00 PM	/	/	902	20.71
6:00 AM	11:00 AM	/	/	359	8.24
11:00 AM	6:00 PM	/	/	234	5.37
10:00 AM	6:00 PM	/	/	177	4.06
2:00 PM	6:00 PM	/	/	114	2.62
6:00 AM	1:00 PM	/	/	96	2.20
8:00 AM	1:00 PM	/	/	80	1.84
8:00 AM	12:00 PM	/	/	75	1.72
			TOTAL=	3277	75.23

(solution quality). Assessing the quality of a solution is not trivial since a number of metrics are of interest, e.g., whether or not a feasible delivery schedule can be produced on a particular day, the travel time of all routes, the duration of all routes, the number of vehicles used, and the number of customers visited by their primary vehicle. Of primary importance to the distributor is the ability to produce a feasible delivery schedule each day.

2.6.1 Impact of parameter choices

We study the impact of parameter choices on the performance of the methodology in this section. For this study, we establish a set of default parameter values (based on initial experimentation) and then vary a single parameter value to assess impact. The default parameter values used were: $\alpha = 0.90$, $\beta = 0.85$, $N = 1000$, and $\gamma = 15$ degrees. For this study, we use data for Thursdays since this is typically the busiest delivery day. Historically, an average of 43 vehicles were used to serve customers on Thursdays and therefore we set $m = 43$.

Time feasibility requirement parameter. The value of the time feasibility requirement parameter β is certainly one of the most important. Since we have the flexibility provided by backup vehicles at the operational level, it should not be necessary to enforce high levels of primary route time feasibility (e.g., $\beta > 0.95$). Note that for a fixed fleet size, higher values of β may lead to infeasible solutions (i.e., not all customers can be accommodated) or higher solution costs, while lower values of β risk operational feasibility.

We investigated performance using the following β values: 0.5, 0.7, 0.85, and 0.95. The results for the primary routes planning heuristic using these values are summarized in Table 3, where we present the run time (in hours), the average time feasibility of the individual primary routes constructed, and the final number of routes.

Table 3: Primary Routes Heuristic Performance for Different Time Feasibility Parameter β Values

β	Run Time (hours)	Avg. Time Feasibility	Number of Primary Routes
0.5	3.01	0.536	39
0.7	3.11	0.736	40
0.85	3.05	0.849	42
0.95	2.95	0.934	42

Not surprisingly, the average time feasibility of the constructed fixed routes is strongly correlated with the choice of the parameter value of β. Furthermore, with a lower time feasibility requirement more routes can be eliminated in the vehicle reduction step. There is little variation in running time. Note additionally that when $\beta = 0.85$ and $\beta = 0.95$, the average time feasibility in the generated routes (0.849 and 0.934 respectively) is less than the target minimum. This is of course possible, since the heuristic includes a phase where β is decreased until all customers are inserted onto fixed routes.

Of course, the average time feasibility of the fixed routes only provides an indication of their operational usefulness. We next evaluate the quality of the daily delivery routes that result from these primary routes. We first determine backup routes using the following parameters: sample size $N_2 = 500$ days, and 300 improvement iterations. The backup route generation results are summarized in Table 4, where we report the number of sample days in which a feasible set of daily delivery routes could not be found, the number of days in which at least one of the irregular customers cannot be inserted, the run time (in minutes), and the average number of iterations required to recover feasibility (if possible).

The results indicate that at this stage, for $\beta = 0.5$, there is one day for which it was not possible to find a feasible delivery plan for the regular customers, and there are five days for which it was not possible to find a delivery plan that serves the regular and the irregular customers. For the other values of β considered, a feasible operational outcome is likely for

Table 4: Backup Route Heuristic Results for Different Time Feasibility Parameter Values

β	0.5	0.7	0.85	0.95
Number of days where feasibility cannot be recovered	1	0	0	0
Number of days where an irregular customer cannot be inserted	5	0	0	0
Run time (minutes)	38.18	30.93	28.06	27.25
Average number of iterations required to recover feasibility	103.31	50.73	20.14	8.16

all realizations. The run time does not seem to be impacted and is about 30 minutes in all cases.

Given backup route assignments, we now test the quality of the planned routes by measuring average daily delivery route performance. For twelve consecutive Thursdays, we used the proposed heuristic to create actual delivery schedules for the realized demands on these days. The results are summarized in Table 5.

Table 5: Daily Route Results for Different Time Feasibility Parameter Values

β	0.5	0.7	0.85	0.95
Number of infeasible days	11	3	0	0
Avg. travel Time	4217	4222	4091	4192
Avg. number of vehicles	38.58	39.00	39.75	39.50
Max. number of vehicles	39	40	42	41
Percentage of customers visited by primary vehicle	64%	61%	61%	61%
Run time (secs)	60.58	59.58	56.75	55.67

Several observations can be made. First, it appears that a β value too small leads to infeasible primary and backup routes. For value $\beta = 0.5$, we are unable to construct a feasible delivery schedule in eleven out of the twelve days, and for $\beta = 0.7$, we are unable to construct a feasible delivery schedule for three of the twelve days. Looking at the results for $\beta = 0.85$ and $\beta = 0.95$, note that the average travel time when $\beta = 0.85$ is slightly lower than when $\beta = 0.95$. Counter to intuition, the average and maximum number of vehicles are slightly higher. Note also that the final solution exploits the flexibility provided by the backup routes, which visit about 39% of the customers. Finally, run times are all in the 50 to 60 second range for daily delivery route construction, and appear to be independent of β. From this analysis, we conclude that β values of 0.85 or 0.95 are appropriate.

Sample size parameter N. As noted earlier, we employ sampling to assess primary route feasibility and quality. The size N of the sample should be chosen to balance a trade-off between the accuracy of this assessment and computational efficiency. In the next set of experiments, we explore this trade-off. Using default values for α, β, and γ, we investigate sample sizes 500, 1000, 2000, and 3000. The results are presented in Table 6.

Table 6: Primary Routes Heuristic Performance Results for Different Sample Size Parameter Values

N	Run Time (hours)	Avg. Time Feasibility	Final Number of Routes
500	1.59	0.827	43
1000	3.05	0.849	42
2000	5.97	0.853	42
3000	9.07	0.857	41

As expected, the impact of the sample size on run times is significant. With $N = 500$, the run time is less than two hours, but with $N = 3000$ the run time is about 9 hours. On the other hand, we see small increases in the average time feasibility of the routes generated. Also, we could eliminate more routes using $N = 3000$. To assess the true quality of these primary routes, we create backup routes using the backup route heuristic (with the parameters specified earlier) starting with each of these sets of primary routes. Table 7 presents these backup routes results.

Note that the average number of iterations required to recover feasibility is slightly higher for smaller sample sizes, but the differences are not significant. Determining backup routes takes the same time in all cases; about 30 minutes.

Table 7: Backup Route Heuristic Results for Different Sample Size Parameter Values

N	500	1000	2000	3000
Number of days where feasibility cannot be recovered	0	0	0	0
Number of days where an irregular customer cannot be inserted	0	0	0	0
Run time (minutes)	29.63	28.06	27.96	28.40
Average number of iterations required to recover feasibility	21.03	20.14	19.08	19.25

Finally, Table 8 presents statistics for the solutions produced by the daily delivery routes construction heuristic for the realized demands on the twelve Thursdays. The differences for the reported characteristics are relatively small, except for one instance where a feasible schedule cannot be found for the smallest sample size of $N = 500$. One may argue that the highest sample size of 3000 produces slightly better results since both the average and the maximum number of vehicles used are slightly lower than for the other cases, and more customers are served by primary vehicles, but overall the results are comparable.

Table 8: Daily Route Results for Different Sample Size Parameter Values

N	500	1000	2000	3000
Number of infeasible days	1	0	0	0
Avg. travel Time	4249	4091	4251	4102
Avg. number of vehicles	40.25	39.75	39.83	39.67
Max. number of vehicles	42	42	41	41
Percentage of customers visited by primary vehicle	60%	61%	60%	63%
Run time (secs)	63.5	56.75	57.00	63.92

Angular Insertion Limit Parameter γ. To reduce computation time spent on evaluating insertions and to avoid unfortunate insertion decision early in the construction process, we enforce an angular insertion limit that controls the radial spread of the customers served by the same primary route. In the next set of experiments, we explore the impact of the value of this parameter. The values of γ investigated are 5, 15, 30, 45, and 60 degrees. The results are presented in Table 9.

Table 9: Primary Routes Heuristic Performance Results for Different Angular Insertion Limit Parameter Values

γ	Run Time (hours)	Avg. Time Feasibility	Final Number of Routes
5	1.85	0.811	42
15	3.05	0.849	42
30	5.15	0.852	42
45	7.47	0.848	42
60	9.33	0.844	42

As expected, run time increases as the angular insertion limit value increases; more insertion candidates are considered, and evaluating each candidate is computationally intensive. Except for the tightest limit of $\gamma = 5$ degrees, the average time feasibilities are comparable. Table 10 presents the statistics related to the creation of the backup vehicle routes. The only interesting observation that can be made here is that the average number of iterations required to recover feasibility is highest for parameter values 5 and 60. Apparently, a limit of 5 degrees is so restrictive that it is hard to recover feasibility.

Table 10: Backup Routes Heuristic Results for Different Angular Insertion Limit Parameter Values

γ	5	15	30	45	60
Number of days where feasibility cannot be recovered	0	0	0	0	0
Number of days where an irregular customer cannot be inserted	0	0	0	0	0
Run time (minutes)	28.33	28.06	27.73	27.91	27.52
Average number of iterations required to recover feasibility	27.86	20.14	20.08	20.91	23.33

Finally, Table 11 presents statistics for the solutions produced by the daily delivery routes construction heuristic for the realized demands on the twelve Thursdays. The results show that γ not only impacts the running time of the fixed routes construction heuristic, but also impacts solution quality. When the limit is too strict (5 degrees) or too loose (60 degrees), the result is some days for which a feasible operational solution cannot be generated. When $\gamma = 5$, a few bad primary routes result from the forced insertion of many customers during the feasibility relaxation phase. On the other hand, when $\gamma = 60$, a few bad primary routes result when poor customer insertion choices are made early in the primary route construction heuristic, and these choices are not undone by the local improvement routines. Overall, $\gamma = 15$ degrees appears to generate the best solution quality at a reasonable computational cost.

Table 11: Daily Route Results for Different Angular Insertion Limit Parameter Values

γ	5	15	30	45	60
Number of infeasible days	2	0	0	0	1
Avg. travel Time	4614	4091	4287	4380	4435
Avg. number of vehicles	40.00	39.75	39.75	39.75	40.25
Max. number of vehicles	42	42	41	42	42
Percentage of customers visited by primary vehicle	60%	61%	57%	55%	57%
Run time (secs)	61.42	56.75	59.33	60.33	54.75

We conclude this part of the study by presenting, in Table 12, statistics for the fixed routes created for Thursday with the default parameter values. More specifically, we report the number of customers on the route; the sample average duration, travel time, service time, and waiting time of the route in minutes; and the sample-based likelihood estimate of delivery window feasibility. For simplicity in the table, we use the notation E[.] to indicate a sample average or estimate. The number of customers served on Thursday (i.e., the number of customers with a positive probability of requiring a delivery) is 1786, which includes 139 irregular customers.

Looking at the averages presented in the final row of the table, note that 5% of the duration represents waiting time, 60% represents service time, and 35% represents travel time. Approximately 42 customers are assigned to each primary route, and the average time feasibility is 0.849.

2.6.2 Comparison with Historical Planned Routes

In this section, we compare in some detail the delivery schedules produced by the proposed technology with both the historical routes planned by the distributor. The distributor currently uses a simple but difficult-to-reproduce methodology for producing a delivery schedule: a dispatcher generates initial proposed routes based on geographic customer clusters, and then modifies them until they are roughly feasible. While some attempt is made to have the same driver visit regular customers on their regular delivery day(s), often this is not possible and no systematic backup driver is specified.

Since no algorithms have been proposed to date in the research literature for the construction of fixed routes for problems with customers that have delivery windows, a comparison with alternative algorithms is not possible. Care must be taken when making a comparison with the historically planned routes and drawing conclusions; note that travel time and service time approximations introduce error, and historical dispatch and customer data also contain errors. Nonetheless, we perform such a comparison for twelve Thursdays and twelve Mondays. The performance of the planned primary and backup routes for different demand realizations is assessed by comparing the daily delivery schedules produced to the planned historical routes for the same day.

Thursdays. The results for the twelve Thursdays can be found in Table 13. For each Thursday, there are three rows with results. The first row, labeled .H, presents statistics for the planned historical routes; the second row, labeled .GT, presents statistics for the routes produced by our methodology; and the third row, labeled .FREE, presents statistics for the routes produced by our methodology when we drop the requirement that a customers has to be served on its primary or its backup route. We have included these statistics to assess the price the company is paying for serving its regular customers with at most two different drivers on a given weekday. The statistics reported are the number of routes, the number of feasible routes, the number of customers served, the number of customers served on their primary routes, the number of customers served on their backup routes, the total miles, the total travel time, the percent improvement over history in terms of miles, and the percent improvement in terms of travel time. Note that the number of customers served on their primary route and the number of customers served on their backup routes do not add up to the total number of customers served because there are a number of irregular customers each day, each of which may be served on any route.

Several important observations can be made. First, the number of routes in the proposed delivery schedules is smaller than the number of routes in the planned historical delivery schedules; the proposed methodology appears to be able to increase vehicle utilization. Second, about three quarters of the planned historical routes are, according to our evaluation, infeasible. In all cases, the reason for the infeasibility is one or more violated delivery windows. In most cases, the vehicle arrives too late at the last or second to last customer on the route. This may be due to differences in travel time approximations and service time approximations. However, in several cases, the vehicle arrives too late already at customers in the middle part of the route. This is more troublesome since these violations cannot be attributed simply to differences in travel time approximations and service time approximations. Third, the proposed delivery schedules substantially reduce the number of miles and the travel time. This suggests that the primary and backup routing methodology provides the necessary flexibility to create low-cost delivery routes for different demand realizations, and that these opportunities may not be simple to find via manual route construction. Fourth, the daily delivery route construction heuristic clearly exploits the flexibility provided by the backup routes, since approximately one third of the customers are visited by their backup vehicle. Finally, the price the company pays for serving its regular customers on a given weekday with at most two different drivers seems acceptable; a reduction in miles of less than 4% can be obtained by dropping this requirement.

Mondays. A similar experiment was conducted for twelve consecutive Mondays. The results are presented in Table 14. The improvements are much more significant on these low-demand days. The proposed delivery schedules contain far fewer routes than the planned historical delivery schedules. It is also clear that there is less need to rely on the backup vehicles to create low-cost delivery schedules; about 25% of the customers are visited on their backup routes. On the other hand, the price the company pays for serving its regular customers with at most two different drivers is much higher in this case; a reduction in miles of more than 8% is possible by dropping this requirement.

The computational effort required by our proposed methodology is much smaller for Monday problems than for Thursdays, due simply to the smaller set of customers that place orders. The primary routes heuristic requires only 17 minutes of computation time (about 10% of the time required for Thursday), and backup route generation requires less than 3 minutes. The daily route generation heuristic requires about 8 seconds on average for Mondays.

2.7 Directions for Future Research

In this chapter, we have proposed a practical fixed routing system that can be applied in settings with high customer demand variability and delivery window constraints. We have introduced two key ideas; a new recourse strategy in which customers can be assigned to *two* planned routes and the use of sampling-based techniques to handle the presence of delivery time window constraints during the construction of planned routes.

Our computational study has shown that the use of fixed delivery routes with the use of backup routes as a recourse strategy can be quite effective even in environments where there is considerable demand variability on a day-to-day basis. The approach presented for constructing primary routes and an associated set of backup routes proceeds in two phases. First, the primary routes are constructed. Second, given a set of primary routes, a set of backup routes is constructed. An alternative to this sequential construction is a method that jointly plans primary and backup vehicle assignments for customers, and a well-designed approach may lead to higher quality results. However, it is not clear how to design such an approach.

From an operational perspective, and perhaps even from an optimization perspective, it is desirable to have balanced fixed delivery routes, i.e., a set of delivery routes that are balanced in terms of the expected quantity of product delivered on the routes and in terms of the expected duration of the routes. However, incorporating such considerations into the methodology is non-trivial. The route elimination procedure is a small step in that direction.

Many of the algorithmic choices were driven by computational requirements of the sample-based route feasibility and quality assessment methodology. An important avenue of research is to investigate whether computational improvements can be found in these steps that do not adversely impact solution quality. We want to briefly illustrate here a few ideas in this direction next:

Sampling Based Approaches to Test Time Feasibility of an Insertion.

Finding more efficient ways of checking time feasibility of an insertion into a fixed route is crucial. Suppose fixed route R has been assigned r customers and we want to find the best insertion position for customer i in route R. There are $r + 1$ possible insertion positions for i as shown in Figure 3; for example case 1 corresponds to inserting i right after depot. If we generate N sample realizations to evaluate each insertion position, then total number of feasibility checks that has to be performed is $N(r + 1)$, which increases as more customers are inserted into route R. In this traditional method, the time feasibility of each insertion position is assessed separately. Instead, we can use a different method which is more efficient. The efficiency comes from the fact that using the same realization, we now evaluate the feasibility of more than one position, and we can also do feasibility check faster. To illustrate these ideas, suppose that we create a sample realization from fixed route R and in that realization the set of customers requesting a delivery are $\{0-1-2-3-4-5-6-0\}$ with depot shown by 0. Let's call this ordered set of customers the *initial case*. Now, we can use the following steps:

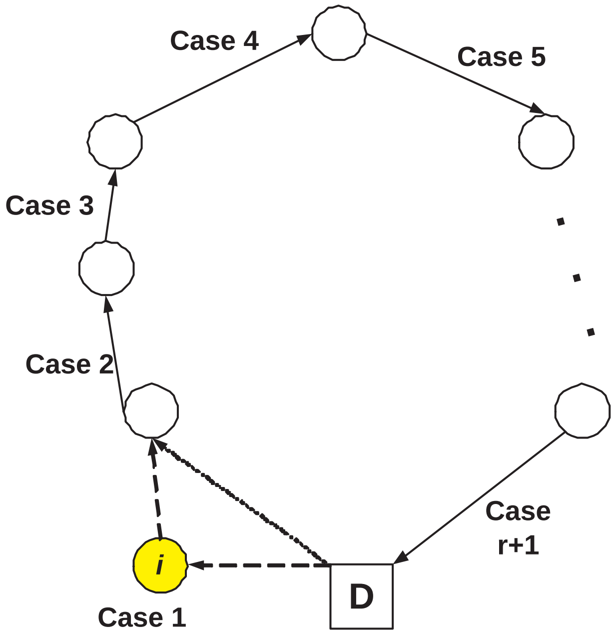

Figure 3: Candidate insertion positions for customer i

1. Evaluate the feasibility of initial case (find arrival, service start, and completion times for each customer). If initial case is infeasible, then stop; insertion of customer i into any position is infeasible too since addition of a customer can only make things worse. Else, go to Step 2.

2. Starting from depot and following the order imposed by Case 1 (i.e. 0-i-1-2-3-4-5-6-0), find the service completion times of customers until you reach a customer after i whose service completion is the same as initial case (note that we may have to check all completion times). If serving customer i or any customer before i is infeasible, then stop, do not consider other remaining cases since they will all be infeasible too. Else, consider the next case and repeat step 2 until all cases are evaluated.

Note that each realization will probably be different, and therefore insertion positions evaluated will also be different from one realization to another. A consensus approach can be adapted to remedy this issue. For example, suppose we have 2 realizations as shown in Figure 4. In realization 1, let the best solution be to insert customer i between customers 3 and 7, while in realization 2, the best place be between customers 5 and 9.

Now, realization 1 does not say anything about where to insert customer i exactly

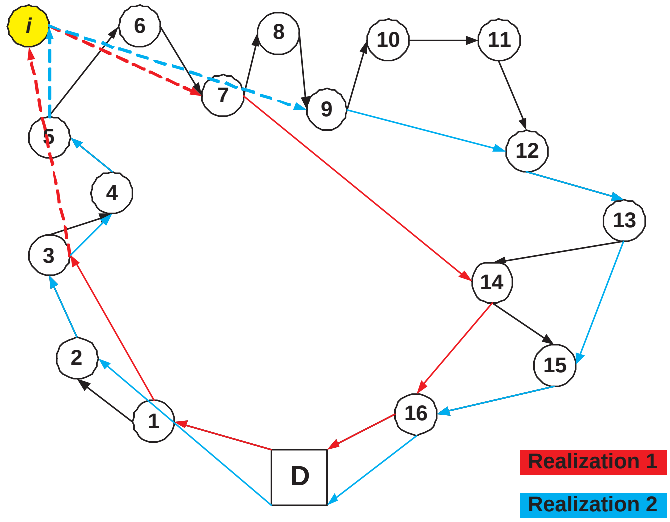

Figure 4: Two realizations of a fixed route and insertion of customer i

between 3 and 7. It can be between 3-4, 4-5, 5-6, or 6-7. Similarly, realization 2 does not differentiate between 5-6, 6-7, 7-8, and 8-9. A frequency based approach can be implemented as follows: Generate N realizations, and for each realization, give one point for best insertion place(s) selected. For example, in realization 1, we would give one point to positions 3-4, 4-5, 5-6, and 6-7. Previously, we required that an insertion position be feasible in at least βN of the realizations. So, this means, in order to accept an insertion position to be feasible, the total accumulated points should exceed βN. If none of the insertion links can accumulate this many points, then insertion of i can be rejected and considered infeasible. Note that in this new method, we always generate N realizations for a fixed route, so even though fixed routes get more crowded, the number of customer realizations evaluated will not change. We believe that developing alternative approaches similar to the one described here is an important area of further research.

In the specific application context that we have studied, there is a set of additional questions that are of interest. In the current setting, we develop primary and backup routes for each weekday separately. By taking this approach, it is clear that a customer may see different drivers if he or she orders on different weekdays from week to week. Alternatively, we may construct fixed and backup delivery routes to be used every day of the week. This will obviously lead to increased delivery costs, but what is the magnitude of this increase? An investigation of other approaches to reduce the number of drivers visiting customers across different weekdays would be interesting. Another interesting, and likely very beneficial, study would be to investigate the cost savings that may result if we can influence the customers' delivery patterns, e.g, by paying an incentive to encourage customers to change their delivery days.

Table 12: Primary Routes Generated with Default Parameter Settings for Thursdays

Route	Customers	E[Duration]	E[Travel Time]	E[Service Time]	E[Wait Time]	E[Time Feasibility]
1	29	601.27	215.80	371.57	13.90	0.852
2	37	701.16	392.23	289.55	19.38	0.878
3	42	655.88	221.83	422.98	11.08	0.850
4	27	697.91	387.98	306.79	3.15	0.909
5	46	590.59	125.45	465.15	0.00	0.725
6	43	725.08	208.73	503.99	12.36	0.895
7	36	672.92	199.63	403.55	69.74	0.861
8	57	538.58	190.16	346.76	1.67	0.856
9	48	608.28	203.93	404.36	0.00	0.750
10	57	719.13	273.97	445.08	0.07	0.714
11	41	610.65	212.98	380.65	17.01	0.866
12	29	408.31	52.17	278.76	77.37	0.886
13	44	588.14	205.43	369.67	13.04	0.858
14	42	637.36	195.81	299.87	141.69	0.860
15	60	491.68	118.08	365.96	7.64	0.787
16	37	468.86	145.77	308.39	14.70	0.932
17	39	672.59	220.87	447.77	3.95	0.850
18	56	515.43	85.33	422.73	7.36	0.862
19	36	611.18	229.18	381.40	0.60	0.983
20	35	605.77	209.10	302.34	94.34	0.873
21	28	530.48	126.75	271.75	131.98	0.865
22	40	484.16	136.19	343.94	4.03	0.900
23	32	696.06	319.80	375.66	0.61	0.891
24	35	528.22	229.69	283.59	14.93	0.818
25	50	603.10	162.33	430.87	9.90	0.879
26	31	428.88	105.54	286.37	36.97	0.878
27	75	633.70	255.38	360.47	17.85	0.852
28	51	731.54	338.61	367.30	25.63	0.618
29	47	697.56	383.47	310.70	3.40	0.857
30	34	732.56	369.06	363.50	0.00	0.875
31	42	712.25	216.41	495.37	0.48	0.857
32	57	693.22	264.54	413.39	15.30	0.856
33	32	570.36	193.74	375.38	1.25	0.874
34	49	593.16	141.81	447.06	4.29	0.760
35	41	526.16	135.96	258.80	131.41	0.939
36	48	729.19	367.28	356.76	5.15	0.851
37	45	472.92	157.23	314.82	0.87	0.898
38	40	610.30	248.19	299.48	62.62	0.852
39	41	682.61	306.37	360.68	15.55	0.894
40	42	587.36	196.45	360.59	30.32	0.855
41	43	588.15	225.11	344.84	18.20	0.775
42	42	614.27	188.29	424.54	1.45	0.782
Average	42.52	608.74	218.16	365.79	24.79	0.849

Table 13: Comparison of Historical Planned and Unrestricted Routes with Proposed Routes for Twelve Thursdays

Day	Routes	Feasible	Customers	Primary	Backup	None	Total Miles	Travel Min	% in Miles	% in Min
1.H	43	11	856	/	/	/	5360	8428	/	/
1.GT	39	39	856	554	241	61	3848	6182	28.2	26.7
1.FREE	32	32	856	379	194	283	3821	6135	28.7	27.2
2.H	42	11	863	/	/	/	5245	8240	/	/
2.GT	38	38	863	536	260	67	3969	6329	24.3	23.2
2.FREE	33	33	863	378	149	336	3858	5649	26.5	31.4
3.H	43	9	876	/	/	/	5638	8805	/	/
3.GT	42	42	876	534	282	60	4194	6686	25.6	24.1
3.FREE	35	35	876	388	213	275	3825	5978	32.2	32.1
4.H	42	6	904	/	/	/	5587	8752	/	/
4.GT	41	41	904	532	300	72	4098	6513	26.7	25.6
4.FREE	33	33	904	416	187	301	3891	5530	30.4	36.8
5.H	42	9	839	/	/	/	5195	8120	/	/
5.GT	41	41	839	516	268	55	3980	6294	23.4	22.5
5.FREE	33	33	839	412	164	263	3907	5607	24.8	31.0
6.H	43	14	879	/	/	/	5410	8491	/	/
6.GT	39	39	879	541	275	63	3900	6255	27.9	26.3
6.FREE	32	32	879	359	157	363	3803	5404	29.7	36.4
7.H	43	14	892	/	/	/	5434	8506	/	/
7.GT	41	41	892	545	294	53	4224	6639	22.3	21.9
7.FREE	33	33	892	423	163	306	4057	5655	25.3	33.5
8.H	43	9	896	/	/	/	5390	8445	/	/
8.GT	41	41	896	551	287	58	4296	6805	20.3	19.4
8.FREE	33	33	896	361	186	349	4055	5501	24.8	34.9
9.H	43	15	815	/	/	/	5245	8177	/	/
9.GT	38	38	815	497	270	48	4184	6579	20.2	19.5
9.FREE	30	30	815	315	173	327	4039	5336	23.0	34.7
10.H	43	12	881	/	/	/	5708	8871	/	/
10.GT	39	39	881	542	273	66	4298	6786	24.7	23.5
10.FREE	32	32	881	438	174	269	4074	5855	28.6	34.0
11.H	42	11	825	/	/	/	5354	8313	/	/
11.GT	39	39	825	489	282	54	3870	6087	27.7	26.8
11.FREE	30	30	825	342	164	319	3814	5011	28.8	39.7
12.H	44	12	853	/	/	/	5757	8960	/	/
12.GT	39	39	853	487	292	74	4232	6671	26.5	25.5
12.FREE	32	32	853	373	169	311	4172	5357	27.5	40.2

Table 14: Comparison of Historical Planned and Unrestricted Routes with Proposed Routes for Twelve Mondays

Day	Routes	Feasible	Customers	Primary	Backup	None	Total Miles	Travel Min	% in Miles	% in Min
1.H	32	28	189	0	0	/	4336	5963	/	/
1.GT	18	18	189	112	43	34	2538	3698	41.5	38.0
1.FREE	15	15	189	92	32	65	2505	3621	42.2	39.3
2.H	32	28	215	0	0	/	4263	5974	/	/
2.GT	19	19	215	123	48	44	2651	3925	37.8	34.3
2.FREE	17	17	215	97	35	83	2619	3830	38.6	35.9
3.H	34	32	214	0	0	/	4534	6285	/	/
3.GT	20	20	214	140	38	36	3000	4342	33.8	30.9
3.FREE	17	17	214	107	33	74	2576	3784	43.2	39.8
4.H	32	31	200	0	0	/	4012	5601	/	/
4.GT	19	19	200	104	57	39	2608	3807	35.0	32.0
4.FREE	16	16	200	84	33	83	2398	3523	40.2	37.1
5.H	32	29	221	0	0	/	4053	5723	/	/
5.GT	19	19	221	130	50	41	2916	4261	28.0	25.5
5.FREE	16	16	221	106	34	81	2640	3739	34.9	34.7
6.H	33	32	202	0	0	/	4514	6232	/	/
6.GT	19	19	202	125	45	32	2709	3928	40.0	37.0
6.FREE	16	16	202	100	31	71	2461	3603	45.5	42.2
7.H	30	29	186	0	0	/	4091	5650	/	/
7.GT	18	18	186	113	40	33	2667	3843	34.8	32.0
7.FREE	16	16	186	91	31	64	2391	3317	41.6	41.3
8.H	31	29	161	0	0	/	4143	5656	/	/
8.GT	17	17	161	92	42	27	2302	3348	44.4	40.8
8.FREE	14	14	161	64	36	61	2290	3333	44.7	41.1
9.H	31	27	200	0	0	/	4138	5743	/	/
9.GT	20	20	200	114	58	28	2621	3826	36.7	33.4
9.FREE	15	15	200	87	40	73	2399	3526	42.0	38.6
10.H	31	30	197	0	0	/	3929	5444	/	/
10.GT	17	17	197	115	45	37	2643	3794	32.7	30.3
10.FREE	14	14	197	92	31	74	2369	3461	39.7	36.4
11.H	33	32	209	0	0	/	4584	6303	/	/
11.GT	20	20	209	131	42	36	2898	4190	36.8	33.5
11.FREE	16	16	209	111	33	65	2737	3781	40.3	40.0
12.H	29	27	172	0	0	/	4031	5543	/	/
12.GT	17	17	172	114	32	26	2535	3651	37.1	34.1
12.FREE	14	14	172	80	22	70	2288	3331	43.2	39.9

CHAPTER III

FIXED ROUTES PROBLEM ON THE INTERVAL

3.1 Introduction

In this chapter, we investigate the VRPSC in a simplified setting in an attempt to develop insights about the value of vehicle sharing strategies. In the simplified setting, customers are assumed to be distributed on the real interval $[0, 1]$, with the depot located at the origin. Such a restriction eliminates the difficult operational vehicle routing problem, which is already NP-hard. Furthermore, we eliminate any operational difficulties that result from packing considerations by limiting the study to problems with known homogeneous customer demands. Finally, we do not include time windows or duration constraints in the study. In this setting, we attempt to understand the value of recourse policies that allow at most two vehicles to share responsibility for a customer, and draw a series of interesting results.

To do so, we study both a traditional fixed routes model in which each customer is assigned to a single fixed route, and a fixed routes with backup vehicles model in which each customer can be assigned to at most two *a priori* vehicle routes. In both cases, we assume an information model where the customers that require service on a given day are known prior to vehicle dispatch. The objective function for both problems is the minimization of the expected operational travel cost. For the traditional fixed routes model, operational routes are found by simply skipping absent customers, so they can be found trivially. However, in the model with backup vehicles, the structure of the recourse policy requires the solution of a new recourse (or second stage) problem that assigns each realized customer to one of its two *a priori* vehicles and then determines routes for all vehicles.

As stated in the literature review in Section 1.5, it is typical in an *a priori* optimization approach for the FRP to design fixed routes that need not be feasible for every customer realization when using a simple customer-skipping operational strategy. Two different approaches are used to ensure that individual fixed routes are not overloaded with customer demand. The first is a chance-constrained approach, in which the probability that each individual fixed route yields an operational route with demand that does not exceed capacity is constrained to be sufficiently large. Chance constraints can be used with an objective function that minimizes the total cost of all fixed routes, or weighted to appropriately capture the total expected costs of all fixed routes operated using the customer-skipping strategy. The second approach is to use a two-stage recourse model, in which a planned *a priori* solution is constructed by a first-stage model that assumes a second-stage recourse problem is solved that recovers feasibility for all instances. Typically, recourse policies are analyzed that assume that each vehicle operates independently in the second stage, and uses a simple detour-to-depot strategy to address capacity failures when many customers arrive on a given day. The objective of these two-stage models is to determine a first-stage solution that minimizes the expected operating cost, given a second-stage recourse problem.

In this chapter, we study two different operational policies for our simplified VRPSC problem. For the traditional fixed routes model, we adopt approach similar to chance-constrained models where we require the *a priori* routes for each vehicle to be capacity feasible for all realizations; note that by doing so, we also ensure implicitly that the maximum duration of each vehicle's route is reasonable. In the fixed routes with backup vehicles model, we instead consider a case with a second-stage recourse problem that assigns customers to either their primary or backup vehicle in order to minimize total operating costs. Note that the first stage problem in this case constrains the assignment of customers to *a priori* routes such that a feasible solution can always be found to this recourse problem.

The remainder of the chapter is organized as follows. We formally define the problem we will study in Section 3.2. Section 3.3 presents results for the traditional fixed routes model in this setting, whereas Section 3.4 extends the study to the new recourse policy that uses backup vehicles. Section 3.5 presents results comparing the different solutions, and Section 3.6 describes future research directions.

3.2 Problem Definition

We consider the following distribution problem on the interval $[0, 1]$. A distribution center is located at 0. A fixed fleet of homogeneous vehicles of capacity Q is available every day to serve n customers. Customer i ($i \in \{1, ..., n\}$) is located at distance $0 < d_i \leq 1$; without loss of generality we assume $d_{i+1} \leq d_i$ for $i = 1, ..., n-1$. On any given day, customer i ($i \in \{1, ..., n\}$) places an order with probability p_i; in case a customer places an order, the order quantity is assumed to be 1 although the results would hold if $q_i = q$ for i ($i \in \{1, ..., n\}$).

In the traditional Fixed Routes Problem, which we study in Section 3.3, the goal is to find a set of routes that can be used to serve daily demand realizations and that minimizes the expected delivery costs. In this setting each customer is assigned to exactly one fixed route, and note that in our problem variant, we assume that the set of fixed routes always yields a feasible solution for every customer realization. In the Fixed Routes Problem with Backup Vehicles, introduced in Erera et al. [25], the goal is to find a set of primary routes *and* a set of backup routes that can be used to serve daily demand realizations and that minimizes the expected delivery costs. Each customer appears on exactly one of the primary routes and on exactly one of the backup routes. Again, we assume that any feasible set of primary and backup routes is such that a feasible set of operational routes serving all customers can be generated for every customer realization. We study this problem in Section 3.4. We further assume that customers do not have time windows and travel cost is proportional to the linear distance traveled.

3.3 Traditional Fixed Routes Problem

In the traditional Fixed Routes Problem, we assign each customer to a single vehicle to minimize the expected total cost. Due to the vehicle capacity restriction, and since an assignment is only feasible if all possible customer realizations can be served feasibly, we can assign at most Q customers to each vehicle.

Note that if customers $\{1', 2', ..., j', ..., Q'\}$ are assigned to the same vehicle, assuming $d_{1'} \geq d_{2'} \geq ... \geq d_{Q'}$, the expected cost associated with this vehicle is $p_{1'}d_{1'} + p_{2'}d_{2'}(1 -$

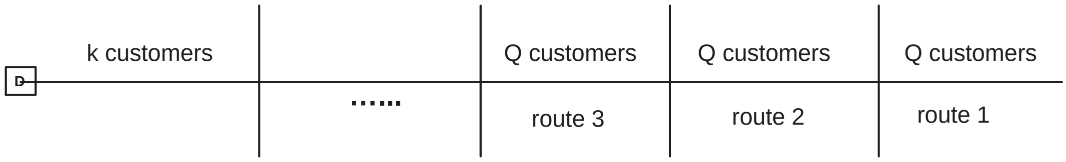

Figure 5: Optimal fixed routes solution for monotone non-decreasing probabilities

$p_{1'}) + ... + p_{j'} d_{j'} \prod_{k=1'}^{(j-1)'} (1 - p_k) + ... + p_{Q'} d_{Q'} \prod_{k=1'}^{(Q-1)'} (1 - p_k)$. We first give the definition of a contiguous assignment (solution) that will be helpful in the remainder of this chapter.

Definition 1 (Contiguous Solution) *Let d_i^{min} and d_i^{max} be the distances of closest and furthest customer served by vehicle i in a given solution. A solution is said to be contiguous if for any pair of vehicles i and j in the solution, we have either $d_i^{min} \geq d_j^{max}$ or $d_j^{min} \geq d_i^{max}$.*

Theorem 2 *The solution given in Figure 5 is optimal for the case $p_1 \geq p_2 \geq ... \geq p_n > 0$.*

Proof. Given any solution, we concentrate on a pair of vehicles at a time. We show that we can decrease the expected cost of the pair by converting it into a contiguous solution. Note that all other vehicles are not affected by this change. Once all pairs of vehicles have been considered and converted, we end up with the solution given in Figure 5.

Consider a solution with more than one vehicle. Suppose there are a customers in one vehicle and b customers in another in that solution and without loss of generality assume $max(a, b) = a$. If $a + b \leq Q$, then all $a + b$ customers can be merged into one vehicle with smaller cost. So, assume $a + b > Q$. Let c_i be the marginal cost associated with customer i, so that total cost is $\sum_{i=1}^{a+b} c_i$. Note that $c_i = p_i d_i \prod_{j \subseteq \{1,2,...,i-1\}} (1 - p_i)$, where multiplication is over all customers who are served in the same vehicle with customer i.

We claim the contiguous solution where furthest a customers are served by the same vehicle is better than all other solutions as given in Figure 6. Here all customers with the same color is served by the same vehicle.

If c_i^x is the marginal cost of customer i in solution x, then we have $c_i^2 \leq c_i^1$ for $1 \leq i \leq a$, and $c_i^2 \geq c_i^1$ for $a + 1 \leq i \leq a + b$ so that we incur smaller marginal costs in Solution 2 for

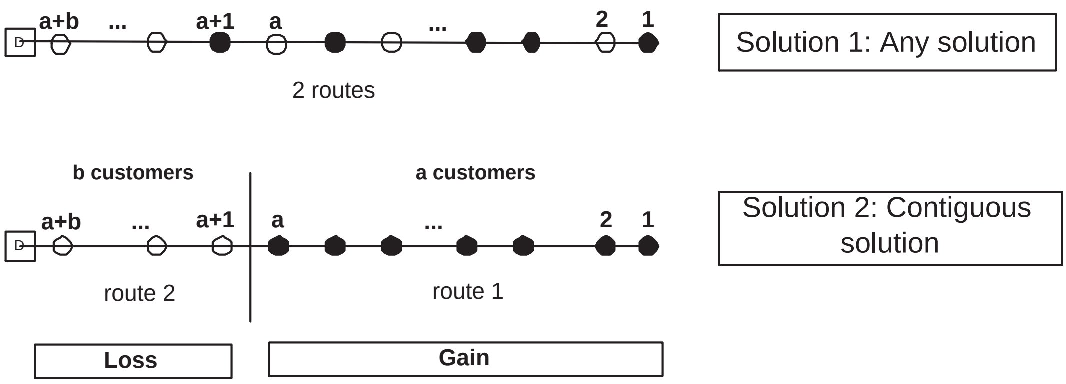

Figure 6: The contiguous vs. non-contiguous solution

customers 1 through a, but larger marginal costs for customers $a+1$ through $a+b$. We compare these gains with losses in this proof.

Let $Gain = \sum_{i=1}^{a}(c_i^1 - c_i^2)$ and $Loss = \sum_{i=a+1}^{a+b}(c_i^2 - c_i^1)$ so that $Gain - Loss = E[Solution1] - E[Solution2]$. Opening up the c_i^2 terms, we can rewrite as follows:

$Gain = [c_1 - p_1 d_1] + [c_2 - p_2 d_2(1-p_1)] + ... + [c_a - p_a d_a \prod_{k=1}^{a-1}(1-p_k)]$.

$Loss = [p_{a+1}d_{a+1} - c_{a+1}] + [p_{a+2}d_{a+2}(1-p_{a+1}) - c_{a+2}] + ... + [p_{a+b}d_{a+b} \prod_{k=a+1}^{a+b-1}(1-p_k) - c_{a+b}]$.

Note that $Gain \geq 0$ and $Loss \geq 0$, and $\frac{\partial Gain}{\partial d_i} \geq 0$ and $\frac{\partial Loss}{\partial d_i} \geq 0$.

For the case where every customer is located at customer a's location (i.e. d_a) and when all probabilities are equal to customer a's probability (i.e. p_a), then we have: $Gain - Loss = E[Solution1] - E[Solution2] = 0$. This can be seen by noting that any solution which assigns a customers to one vehicle and b customers to another vehicle will yield the same cost. We now prove that $\frac{\partial Gain}{\partial p_i} \geq 0$ and $\frac{\partial Loss}{\partial p_i} \geq 0$, so that since we have $d_1 \geq d_2 \geq ... \geq d_a \geq d_{a+1} \geq ... \geq d_{a+b}$ and $p_1 \geq p_2 \geq ... \geq p_a \geq p_{a+1} \geq ... \geq p_{a+b}$, distances and probabilities for gain will increase and those for loss will decrease, which means that $Gain$ will increase and $Loss$ will decrease, keeping $Gain - Loss \geq 0$ true.

Claim 1: $\dfrac{\partial Gain}{\partial p_i} \geq 0$ for $1 \leq i \leq a+b$.

$\frac{\partial Gain}{\partial p_i} = 0$ for $a+1 \leq i \leq a+b$ since it does not include p_i. Let $i = k$ where $1 \leq k \leq a$ as given in Figure 7.

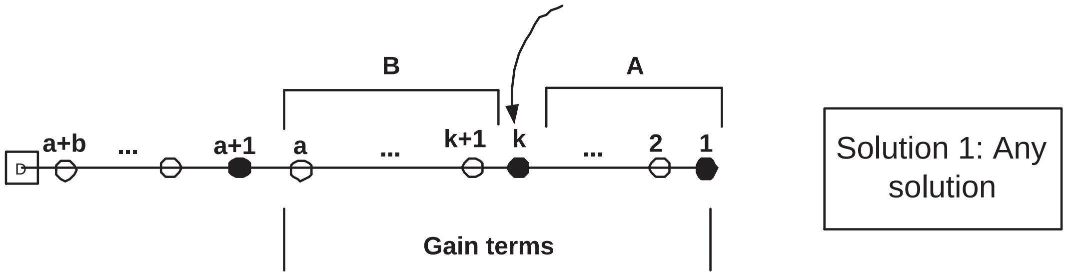

Figure 7: Any given solution on the interval: Gain terms

$$\frac{\partial Gain}{\partial p_k} = \sum_{i=k}^{a} \frac{\partial c_i}{\partial p_k} - d_k \prod_{i=1}^{k-1}(1-p_i) + p_{k+1}d_{k+1} \prod_{i=1, i\neq k}^{k} (1-p_i) + ... + p_a d_a \prod_{i=1, i\neq k}^{a-1} (1-p_i).$$

We will prove that this partial derivative is non-negative for any solution of Figure 7, by showing that its minimum value is nonnegative. In Figure 7, we will select customers from Sets A and B that will be served together by customer k, in a way to minimize the expression for $\frac{\partial Gain}{\partial p_k}$. Since $\sum_{i=k}^{a} \frac{\partial c_i}{\partial p_k}$ is the only non-constant part, we try to minimize it. Note that $c_k = p_k d_k \prod_{j\subseteq A}(1-p_i)$ where product is accumulated over customers that are served by the same vehicle as customer k, so $\frac{\partial c_k}{\partial p_k} = d_k * \prod_{j\subseteq A}(1-p_j)$. For $i \subseteq B$, $\frac{\partial c_i}{\partial p_k} = 0$ if customer i is not served by the same vehicle as customer k, and $\frac{\partial c_i}{\partial p_k} = -p_i d_i \prod_{j\subseteq A\cup\{k+1,...,i-1\}} (1-p_j)$ otherwise. For simplicity, let $M = \sum_{i=k}^{a} \frac{\partial c_i}{\partial p_k}$ and we try to minimize M by selecting customers to be served together with customer k from sets A and B. Let A^* and B^* be the selections which minimize M. We now characterize A^* and B^*. We now provide the following results whose proofs are provided in Appendix. These ideas are illustrated in Figure 8. To minimize M;

R1. For a given selection $B' \subseteq B$, customers in A should be selected in the order of 1, 2, 3, ..., $k-1$.

R2. For a given selection $A' \subseteq A$, customers in B should be selected in the order of $k+1$, $k+2$, ..., a.

R3. First all the customers in A should be selected before any customer in B is selected.

Note that since we try to minimize M, we are better off if customer k is served by the vehicle that serve a customers (remember $a = max\{a, b\}$), so that we can make more

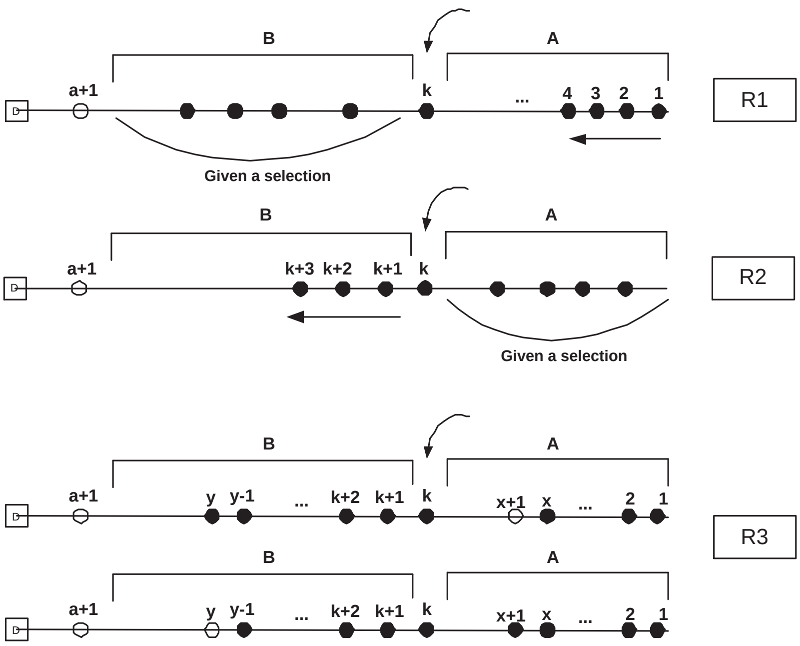

Figure 8: Characterization of selection sets A^* and B^* to minimize gain derivative

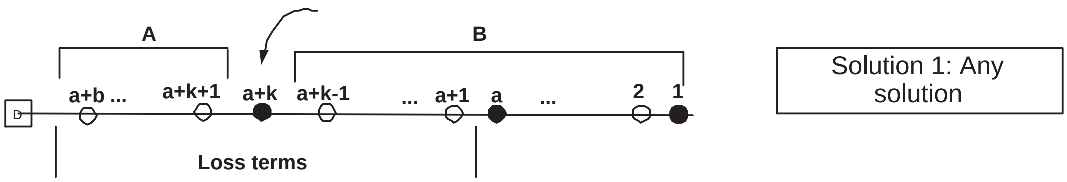

Figure 9: Any given solution on the interval: Loss terms

selections and decrease M. Combining R1, R2, and R3 with this observation, we select customers $\{1, 2, ..., k-1, k+1, k+2, ..., a\}$ to minimize M. Substituting minimum M into partial derivative, we see $\dfrac{\partial Gain}{\partial p_k} = 0$.

Claim 2: $\dfrac{\partial Loss}{\partial p_i} \geq 0$ for $1 \leq i \leq a+b$.

$Loss = [p_{a+1}d_{a+1} + p_{a+2}d_{a+2}(1-p_{a+1}) + ... + p_{a+b}d_{a+b}\prod_{i=a+1}^{a+b-1}(1-p_i)] - [c_{a+1} + c_{a+2} + ... + c_{a+b}]$ with the first part being constant.

For $1 \leq i \leq a, \frac{\partial Loss}{\partial p_i} \geq 0$ since $\frac{\partial c_x}{\partial p_i}$ is either zero or negative, and constant terms do not include p_i. Let $i = a+k$ where $1 \leq k \leq a+b$ as given in Figure 9.

Since the $Loss$ term does not include p_{a+k} for the first $k-1$ terms, we have:

$\frac{\partial Loss}{\partial p_{a+k}} = [d_{a+k} \prod_{i=a+1}^{a+k-1} (1-p_i) - p_{a+k+1} d_{a+k+1} \prod_{i=a+1, i \neq a+k}^{a+k} (1-p_i) - ... - p_{a+b} d_{a+b} \prod_{i=a+1, i \neq a+k}^{a+b-1} (1-p_i)] - [\sum_{i=a+k}^{a+b} \frac{\partial c_i}{\partial p_{a+k}}]$. We try to minimize this expression by making selections from customer sets A and B in Figure 9 that will be served together with customer $a+k$. Let non-constant terms be shown by $H = \sum_{i=a+k}^{a+b} \frac{\partial c_i}{\partial p_{a+k}}$ so that we try to maximize H.

$H = d_{a+k} \prod_{j \subseteq B} (1-p_j) + \sum_{i=a+k+1}^{a+b} \frac{\partial c_i}{\partial p_{a+k}}$. We now provide similar results which characterize the optimal selection from sets A and B to maximize H. The proofs are provided in the Appendix. The ideas are summarized in Figure 10.

S1. Given a selection $A' \subseteq A$, customers in B should be selected in the order of $a+k-1$, $a+k-2$, ...1.

S2. Given a selection $B' \subseteq B$, customers in A should be selected in the order of $a+b$, $a+b-1$, ..., $a+k+1$.

S3. First all the customers in A should be selected before any customer in B is selected.

Note that since partial derivatives become negative if we make more selections from A and $\prod_{j \subseteq B}(1-p_j)$ decreases if we make more selections from B, to maximize H it is better if customer $a+k$ is served by the vehicle who has b customers (Remember that $max\{a,b\} = a$).

Combining S1, S2, and S3 with this observation, we select $\{a+1, a+2, ..., a+k-1, a+k+1, ..., a+b\}$ to maximize H. Substituting maximum H into partial derivative, we see that $\frac{\partial Loss}{\partial p_{a+k}} = 0$.

This analysis shows that Solution 2 given in Figure 6 has a smaller expected cost than Solution 1. Let Solution 3 be defined when customer $a+1$ is also served by route 1. We complete the proof by showing that $E[Cost_3] \leq E[Cost_2]$:

$E[Cost_2] = F(a) + p_{a+1}d_{a+1} + p_{a+2}d_{a+2}(1-p_{a+1}) + ... + p_{a+b}d_{a+b} \prod_{i=a+1}^{a+b-1} (1-p_i)$, where F(a) is the expected cost associated with customers 1 through a.

$E[Cost_3] = F(a) + p_{a+1}d_{a+1} \prod_{i=1}^{a} (1-p_i) + p_{a+2}d_{a+2} + p_{a+3}d_{a+3}(1-p_{a+2}) + ... + p_{a+b}d_{a+b} \prod_{i=a+2}^{a+b-1} (1-p_i)$

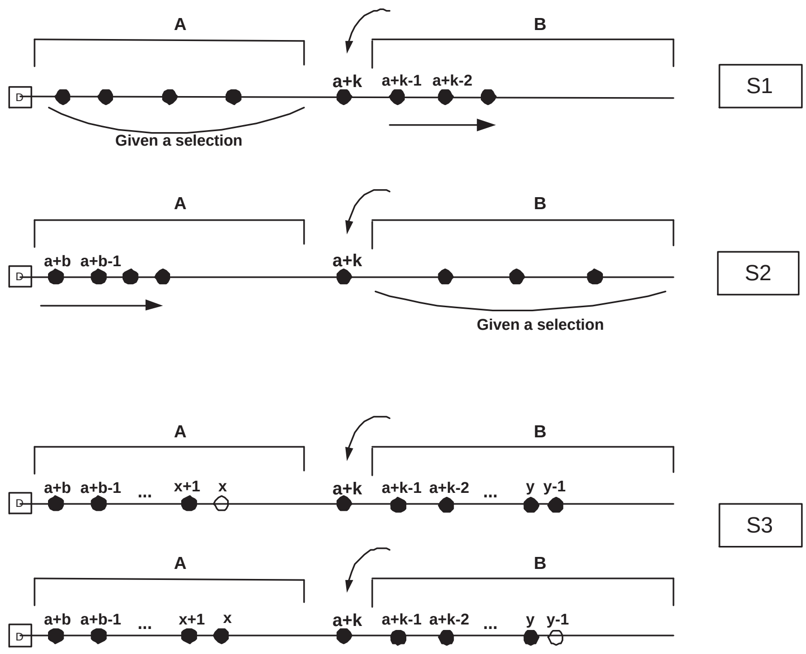

Figure 10: Characterization of selection sets A^* and B^* to minimize loss derivative

letting $\alpha = p_{a+2}d_{a+2} + p_{a+3}d_{a+3}(1-p_{a+2}) + ... + p_{a+b}d_{a+b}\prod_{i=a+2}^{a+b-1}(1-p_i)$, we check if:

$E[Cost_2] - E[Cost_3] = p_{a+1}d_{a+1}[1-\prod_{i=1}^{a}(1-p_i)] - p_{a+1}\alpha \geq 0$. Canceling p_{a+1}, replacing all d_i terms in α with d_{a+1} and canceling d_{a+1}, and using the fact that $p_{a+2} + p_{a+3}(1-p_{a+2}) + ... + p_{a+b}\prod_{i=a+2}^{a+b-1}(1-p_i) = 1 - \prod_{i=a+2}^{a+b}(1-p_i)$, checking $E[Cost_2] - E[Cost_3] \geq 0$ reduces to checking $\prod_{i=1}^{a}(1-p_i) \leq \prod_{i=a+2}^{a+b}(1-p_i)$. Since $a \geq b-1$ the inequality clearly holds because there are a terms on the left hand side, all of which are smaller than $b-1$ terms on the right hand side.

□

Observation 1 *Optimal traditional fixed routes solution cannot be found by a simple swap of two customers.*

Let $n = 8$ customers be located at distances $\{10, 10, 5, 5, 5, 5, 1, 1\}$ respectively with $d_1 = 10$, $d_2 = 10$, $d_3 = 5$, etc., $p_i = 0.1$ for all i, and $Q = 4$. Suppose in the initial solution, customers are assigned to vehicles 1 and 2 as shown in Figure 11: $v_1 = \{1, 2, 7, 8\}$ and

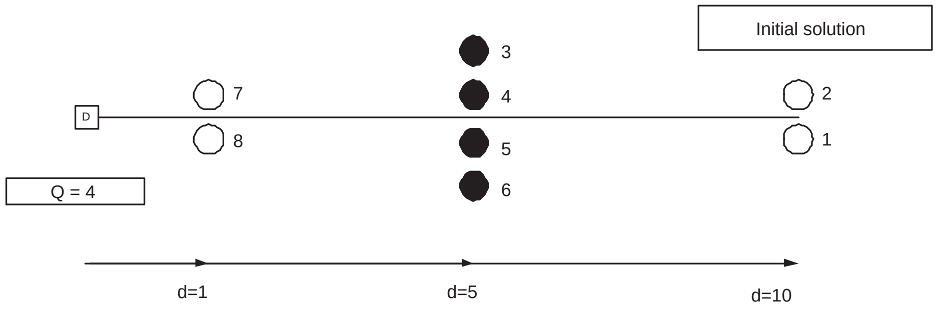

Figure 11: An example solution where single swap does not improve cost

$v_2 = \{3, 4, 5, 6\}$. In this case, $E[Cost_{Initial}] = 3.7734$.

Now, we try swapping the vehicle assignment of two customers served by different vehicles in the initial solution and see whether the expected cost improves. Table 15 shows 16 potential swaps that include one customer from each fixed route and the resulting expected cost of the new solution. As it can be seen, none of the swaps result in an improvement over the initial solution.

Table 15: All possible swaps with corresponding solution costs

Swaps	v_1	v_2	Cost
Swap of 1-3	2,3,7,8	1,4,5,6	3.8234
Swap of 1-4	2,4,7,8	1,3,5,6	3.8234
Swap of 1-5	2,5,7,8	1,3,4,6	3.8234
Swap of 1-6	2,6,7,8	1,3,4,5	3.8234
Swap of 2-3	1,3,7,8	2,4,5,6	3.8234
Swap of 2-4	1,4,7,8	2,3,5,6	3.8234
Swap of 2-5	1,5,7,8	2,3,4,6	3.8234
Swap of 2-6	1,6,7,8	2,3,4,5	3.8234
Swap of 3-7	1,2,3,8	4,5,6,7	3.8058
Swap of 3-8	1,2,3,7	4,5,6,8	3.8058
Swap of 4-7	1,2,4,8	3,5,6,7	3.8058
Swap of 4-8	1,2,4,7	3,5,6,8	3.8058
Swap of 5-7	1,2,5,8	3,4,6,7	3.8058
Swap of 5-8	1,2,5,7	3,4,6,8	3.8058
Swap of 6-7	1,2,6,8	3,4,5,7	3.8058
Swap of 6-8	1,2,6,7	3.4,5,8	3.8058

Observation 2 *Optimal solution is not necessarily of the form given in Figure 5 for the*

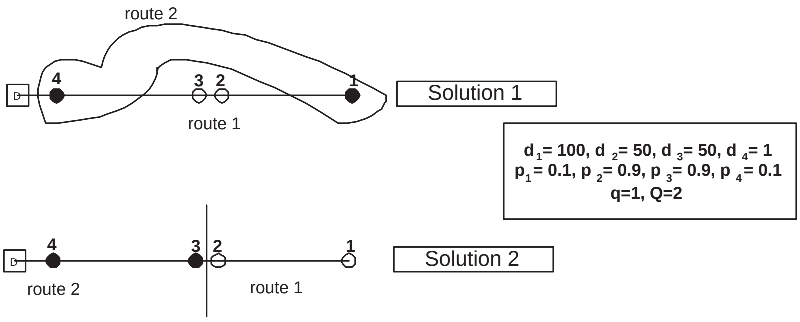

Figure 12: Comparison of two solutions where non-contiguous one is better

general case where $p_i \neq p$ *for each* i*.*

Intuitively, when probabilities are different, there may be situations where assigning another customer into a fixed route will have the effect of dispatching a new vehicle. For example, expected cost of route x may be low due to low probability customers, but when we append another customer with a high probability to vehicle x, then it may increase its expected cost a lot. Figure 12 gives such an example where solution of the form given in Figure 5 gives a larger expected cost as:

$E[solution1] = [0.1 * 100 + 0.9 * 0.1 * 1] + [0.9 * 50 + 0.1 * 0.9 * 50] = 59.59.$

$E[solution2] = [0.1 * 100 + 0.9 * 0.9 * 50] + [0.9 * 50 + 0.1 * 0.1 * 1] = 95.51.$

Observation 3 *Finding minimum number of vehicles necessary for feasibility for the case where* $q_i \neq 1$ *for each* i *is NP complete.*

If $q_i \neq 1$ for all i, the problem of finding the minimum number of vehicles required for feasibility is equivalent to Bin Packing Problem. Therefore, for general order quantities and a given number of vehicles to use, even checking whether we can find a set of feasible fixed routes is difficult.

Note that for different quantities q_i, one might try to adjust First Fit Decreasing heuristic as follows: Starting from the furthest customer, assign a customer to the first route in which it fits. If addition of a customer exceeds vehicle capacity, open a new route. However, as the example in Figure 13 illustrates, this algorithm is not necessarily optimal since,

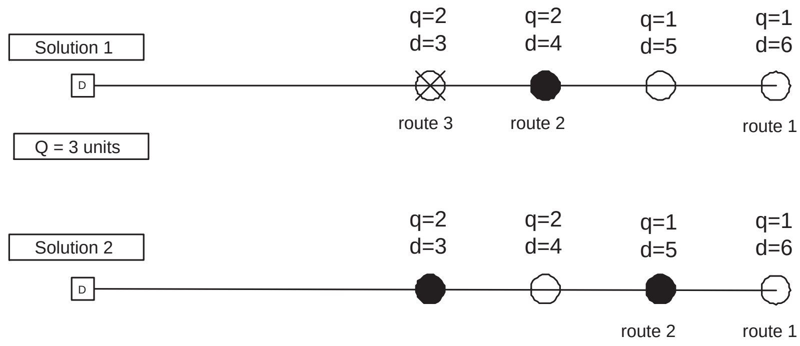

Figure 13: A first fit decreasing heuristic on fixed routes with $p_i = p$ and $q_i \neq 1$ for all i

$E[Cost1] = [12p+10p(1-p)]+[8p]+[6p]$, and $E[Cost2] = [12p+8p(1-p)]+[10p+6p(1-p)]$ so that $E[Cost1] - E[Cost2] = 4p^2 > 0$.

Observation 4 *For $p_i \neq p$, $q_i = 1$ for each i and $Q = 2$, the optimal solution can be found by solving a maximum weighted matching problem in a general (non-bipartite) graph.*

Assume for simplicity that n is even, so that in the optimal solution each vehicle will serve exactly two customers, i.e. n customers are partitioned into exactly $\frac{n}{2}$ pairs in any solution. Let customers i and j with $j > i$ be assigned to vehicle k. Then, expected cost of this vehicle becomes: E[vehicle k] $= p_i d_i + p_j d_j (1 - p_i) = p_i d_i + p_j d_j - p_i p_j d_j$. Since the term $\sum_{(i,j)} p_i d_i + p_j d_j$ will be the same in all solutions, it should be clear that minimizing expected total cost is equivalent to maximizing the expression $\sum_{(i,j)} p_i p_j min\{d_i, d_j\}$, where summation is over all $\frac{n}{2}$ pairs.

We define undirected graph $G = (V, E)$ where $V = \{1, 2, ..., n\}$, $E = \{(i, j), i \neq j, i \in V, j \in V\}$ with weights $w_{ij} = p_i p_j min\{d_i, d_j\}$ for $(i, j) \in E$.

Note that even if n is odd, then we can still find the optimal solution by solving a maximum weighted matching problem in the same G. The node that is not incident to any edge will denote the customer that will be served alone in this case.

There are $O(n^3)$ implementations of non-bipartite weighted matching problems (see Gabow [27]). A faster algorithm for this problem is $O(nm + n^2 \log n)$ due to Gabow [28].

3.4 Fixed Routes with Backup Vehicles

In Section 3.3, we considered traditional fixed routes where there is no sharing of vehicles, i.e. we assign each customer to one of the fixed routes so that the expected total cost of all routes is minimized. In this section, we allow limited vehicle sharing so that each customer can be assigned to two routes. In the Fixed Routes Problem with Backup Vehicles, introduced in Erera et al. [25], the goal is to find a set of primary (fixed) routes *and* a set of backup routes that can be used to serve daily demand realizations and that minimizes the expected delivery costs. Each customer appears on exactly one of the primary routes and on exactly one of the backup routes.

In this section, we investigate properties of the Fixed Routes Problem with Backup Vehicles when customers are assumed to be located on the interval, with depot being at one end. We start with an observation about the feasibility of a given set of primary and backup route solution.

Observation 5 *Let* (x_i, y_i) *be the primary and backup route for customer* i. *For a given planning level solution* (x_i, y_i) *for* i *(*$i \in \{1, ..., n\}$*), its feasibility can be checked by solving a bipartite cardinality matching problem.*

Proof. First note that in the conservative case, it is sufficient to check feasibility of the routes for the demand realization where all customers request a delivery. We create the bipartite graph G as follows:

1. For each customer, we create a node.
2. For each vehicle, we create Q nodes.
3. For each customer i, we create edges between node i and the nodes corresponding to vehicles that can serve customer i.

We solve a Maximum Cardinality Matching Problem on G. If the maximum cardinality found equals n, then the given primary-backup route pairs is feasible, otherwise it is not. 14 illustrates this idea on a simple example.

□

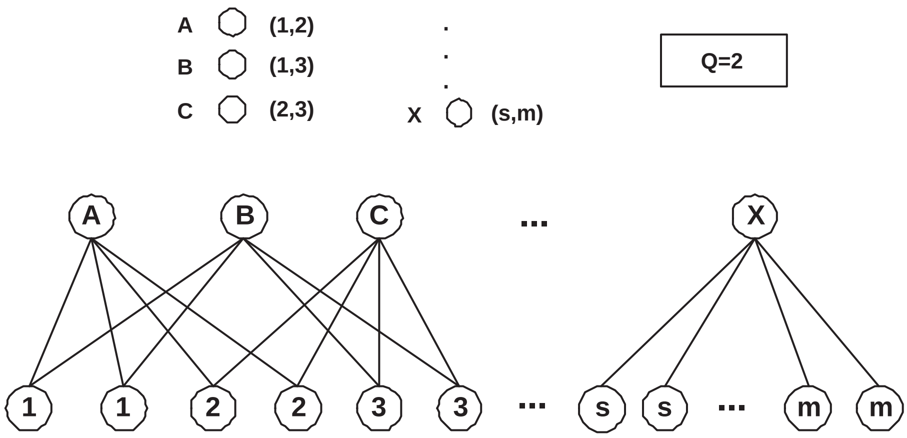

Figure 14: Correspondence between feasibility and maximum cardinality matching

Calculating the operational cost of a vehicle on a particular day, which we assume is equal to the length of the route performed by that vehicle, is trivial as it is simply twice the distance to the farthest customer visited. Therefore, calculating the operational cost for a particular demand realization is trivial in case traditional fixed routes are employed. However, calculating the operational cost for a particular demand realization in case customers can be served on either their fixed route or on their backup route is more difficult. In fact, we have the following theorem.

Theorem 3 . *Minimizing the operational cost for a particular demand realization for a given set of fixed routes and backup routes is NP-complete.*

Proof. The decision version of the operational problem for the fixed routes problem with backup vehicles can be stated as follows. Given a set of vehicles and a set of customers, each with a primary and backup vehicle assigned, does there exist a set of delivery routes serving all customers, respecting the vehicle assignments, and with cost less than or equal to z? NP-completeness is proved by providing a reduction from vertex cover on graphs with a maximum degree of three (shown to be NP-complete by Garey et al. [29]). Given a graph $G = (V, E)$ and an integer k, does there exist a vertex cover of size k?

We create a set of $|V|$ vehicles with capacity 3, each associated with a vertex in the graph. For each edge $\{i, j\} \in E$, we create a customer at 0.5 with primary vehicle i and

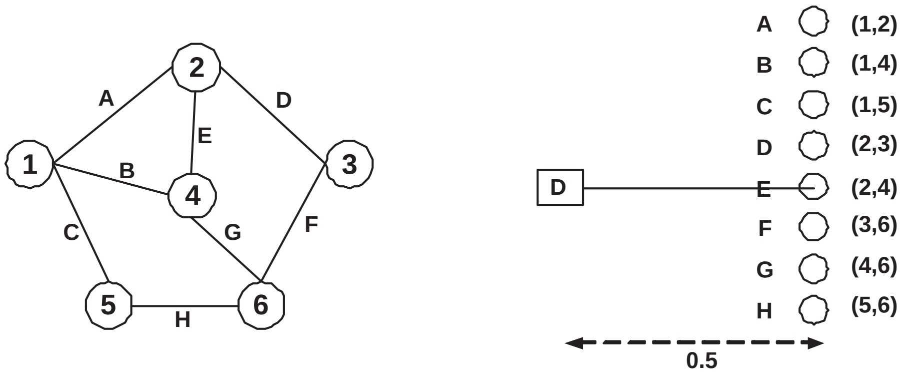

Figure 15: NP-completeness of the operational problem

backup vehicle j. The value z is set to k. Note that since all customers are located at 0.5, the cost of any route is equal to 1. Therefore, the question is whether all customers can be served with k vehicles. Assume that there exists a vertex cover of size k in G. Every edge in G is incident to at least one of the vertices in the cover, which means that the vehicles associated with these vertices can serve all the customers. Thus, there exist a set of delivery routes serving all customers, respecting the vehicle assignments, and with cost less than or equal to k. Similarly, if there exist a set of delivery routes serving all customers, respecting the vehicle assignments, and with cost less than or equal to k, then the vertices associated with the k vehicles form a vertex cover of G. □

Figure 15 illustrates this reduction via an example with eight customers and six vehicles.

Theorem 3 states that operational problem is difficult independent of the difficulty associated with routing problem. Therefore, recourse optimization problem introduced by the limited vehicle sharing is itself not easy to solve.

We next identify a specific class of primary/backup solution pairs (x_i, y_i) where the operational problem becomes easy to solve.

Theorem 4 . *For the class of solutions characterized by $x_i \geq x_j$, and $y_i = x_i + 1$ for any $i > j$, operational problem is easy and can be solved in $O(n^2)$.*

Figure 16 illustrates this specific class of solutions defined by $x_i \geq x_j$, and $y_i = x_i + 1$ for any $i > j$. Note that this type of solution is a natural extension of the contiguous

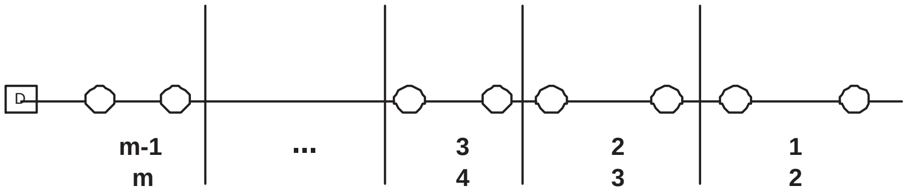

Figure 16: A specific class of solutions for fixed routes with backup vehicles problem

traditional fixed routes. Without loss of generality, we assume that x_i is increasing by 1. This class of solutions can be represented sufficiently by the boundaries as given in Figure 16. In this solution, all the customers in the furthest group can be served either by route 1 or route 2, the second furthest group customers can be served by route 2 or route 3, and so on. Before presenting the proof, we show that feasibility check can be performed faster for this class.

Observation 6 *Feasibility of a planning level solution where $x_i \geq x_j$, and $y_i = x_i + 1$ for any $i > j$ can be checked in time O(n).*

Consider the customer realization where all customers request a delivery. Starting from the furthest customer from depot, assign customers into the lowest indexed available route. Try to fill the lowest indexed route as much as possible. If the lowest indexed route fills up or there are no customers left that can be served by the lowest indexed route, then go to next available lowest indexed route and start filling it up. Continue until either all customers are assigned (in which case the solution is concluded to be feasible) or a customer cannot be served because both of its routes are already full (in which case the solution is concluded to be infeasible).

This approach never wastes route capacity starting from the lowest indexed route and try to fill each route as much as possible. Therefore, if one cannot find a feasible solution by this approach, then any other use of routes will not yield a feasible solution either. Note that we can check feasibility by a single pass over n customers resulting in O(n).

We now describe a simple algorithm which can be used to solve operational problem for the specific class of solutions defined in Theorem 4.

Let $C = \{v_1, v_2, ..., v_k\}$ be the set of customers who request a delivery on a day with $d_{v_i} \geq d_{v_j}$ for $i < j$. Suppose primary and backup routes for customer v_i be shown by (f_{v_i}, b_{v_i}). Let C_j denote the set of customers to be served by route j (i.e. decision variable).

Algorithm 2 Need Check (NC)

> $r = f_{v_1}$ and $C_j = \emptyset$ for all j.
> **while** $C \neq \emptyset$ **do**
> Check if route r is needed for serving customers in C using all non-eliminated routes feasibly.
> **if** r is needed **then**
> Starting from the furthest customer in C, assign as many customers as possible to r, add these assigned customers to C_r. Update $C \leftarrow C \backslash C_r$
> **else**
> Do not assign any customer to r, i.e. $C_r = \emptyset$
> **end if**
> Eliminate r from route lists of customers in C. Update $r \leftarrow r + 1$.
> **end while**

Since we may have n customers in the worst case to serve, we may have to check feasibility for each customer in time O(n) using Observation 6. So, NC can be finished in time O(n^2).

Proof. It should be clear that if we need to use a route for feasibility, we are better off if we use it as much as possible starting from the furthest available customer. The more interesting case occurs when we have the option of not using the lowest indexed available route during the NC algorithm. In this case, we claim that not using the lowest indexed route results in a better solution in terms of total operational cost.

We now present a sketch of the proof. Suppose we are given a specific customer realization, as shown in 17. There are n_1 arriving customers that can be served by routes 1 and 2, n_2 arriving customers with routes 2 and 3, etc. Note that if $n_i = 0$ and $n_j = 0$ for some $i < j$, then decisions regarding customers $n_{i+1}, n_{i+2}, ..., n_{j-1}$ are disconnected from decisions regarding all other remaining customers in the realization. Therefore, WLOG, we assume that $n_i > 0$ for $i = 1, 2, ..., k$ for some k with $n_{k+1} = 0$. For notational convenience, we re-number the customers in the realization from 1 to $\sum_{i=1}^{k} n_i$ starting from the furthest customer.

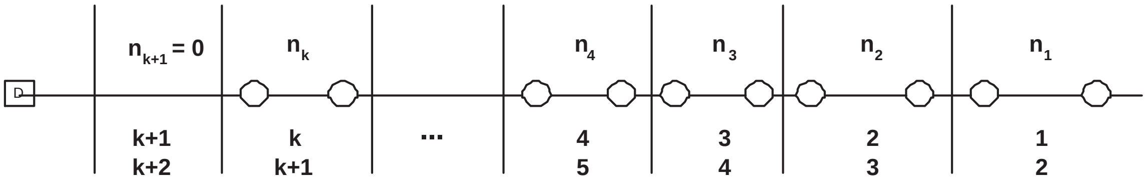

Figure 17: A given customer realization

Suppose that we have the option of not using route 1. This means that we have k inequalities given by $\sum_{i=1}^{s} n_i \leq sQ$ for $s = 1, 2, ..., k$. To illustrate the inequality for $s = 2$, since we do not need route 1 for feasibility, we need to be able to serve $n_1 + n_2$ customers by using routes 2 and 3, hence requiring $n_1 + n_2 \leq 2Q$.

If $n_1+n_2 \leq Q$, NC serves all n_1+n_2 customers by route 2 with a cost of $2d_1$ and will use Routes 3,4,...,k+1 for the remaining customers. Since $2d_1$ is the smallest cost to serve n_1+n_2 customers, NC makes the optimal decision. So, suppose $n_1+n_2 > Q$. If $n_1+n_2+n_3 \leq 2Q$, then NC serves furthest Q customers by route 2 and customers $\{Q+1, Q+2, ..., n_1+n_2+n_3\}$ by route 3, with a cost of $2(d_1+d_{Q+1})$, and will use routes 4,5,..,k+1 for remaining customers. Since we $n_1 + n_2 > Q$, we need at least two routes to serve customers $n_1 + n_2 + n_3$, and $2(d_1 + d_{Q+1})$ is the smallest cost to serve them. Therefore, NC is optimal again. If $n_1+n_2+n_3 > 2Q$, we check whether $n_1+n_2+n_3+n_4 \leq 3Q$. If $n_1+n_2+n_3+n_4 \leq 3Q$, then NC incurs a cost of $2(d_1 + d_{Q+1} + d_{2Q+1})$ to serve customers $n_1 + n_2 + n_2 + n_3 + n_4$, which is again the minimum possible cost to serve since we need at least 3 routes. Continuing in this manner, there will be either a customer group x with $\sum_{i=1}^{s} n_i > (s-1)Q$ for all $s = 1, 2, ..., x-1$ but $\sum_{i=1}^{x} n_i \leq (x-1)Q$ so that NC will make the optimal decision, OR there will be no such group and we will have $\sum_{i=1}^{s} n_i > (s-1)Q$ for $s = 1, 2, ..., k$, in which case NC will incur a cost of $2(d_1 + d_{Q+1} + d_{2Q+1} + ... + d_{(k-2)Q+1} + d_{(k-1)Q+1})$ as shown in Figure 18. Due to inequality $\sum_{i=1}^{k} n_i > (k-1)Q$, we need at least k routes to serve customers, but $2(d_1 + d_{Q+1} + d_{2Q+1} + ... + d_{(k-2)Q+1} + d_{(k-1)Q+1})$ is the minimum possible cost, hence showing the optimality of NC decision.

□

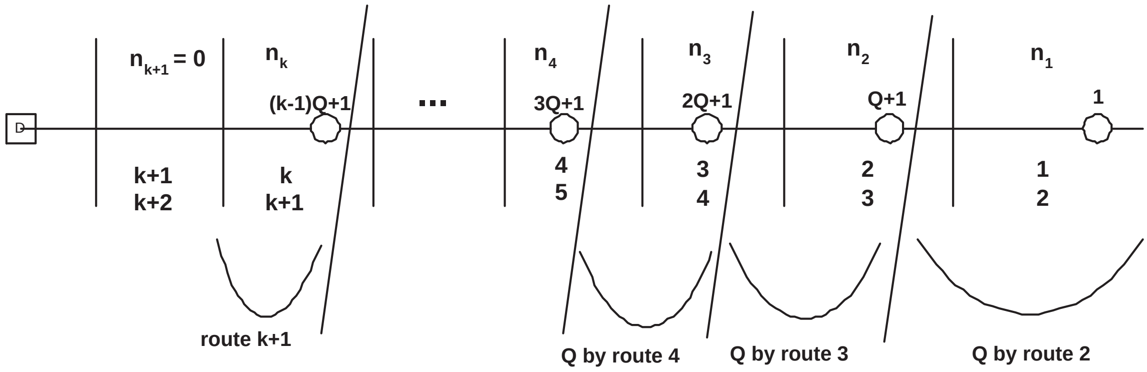

Figure 18: The decisions made by Need Check algorithm

3.5 Comparison of Traditional Fixed Routes and Sharing Policies

In this section, we compare different sharing policies in an effort to characterize the optimum sharing policy in terms of minimum expected total cost. However, since the operational problem is hard to solve for general sharing policies, we restrict ourselves to those planning level solutions for which the operational problem is easy. Figure 19 shows the no sharing policy (traditional fixed routes) and two sharing policies which belong to the class of policies described earlier. Note that sharing policy 0 is a natural extension of no sharing policy, but sharing policy 1 is slightly different. It is clear that sharing policy 0 produces a better operational solution than no sharing policy, because there are additional routes for customers in sharing policy 0 which can only decrease the operational cost.

Proposition 5 *Sharing Policy 1 results in a lower expected cost than Sharing Policy 0, where expectation is taken over all customer realizations.*

Proof. Let X_1 be the number of arriving customers among the furthest Q customers, X_2 be the number of arriving customers among the second furthest Q customers, etc. in Sharing Policy 0 in Figure 19.

In realizations where $X_1 = 0$, it is easy to see that $Cost(Sharing_1) \leq Cost(Sharing_0)$ because Sharing Policy 1 has an additional route, i.e. route m. For realizations where $0 < X_1 < Q$, if route 1 is not needed for feasibility in policy 0, then we do not use it, and we have the following routes to use in both policies as shown in Figure 20. Since policy 1

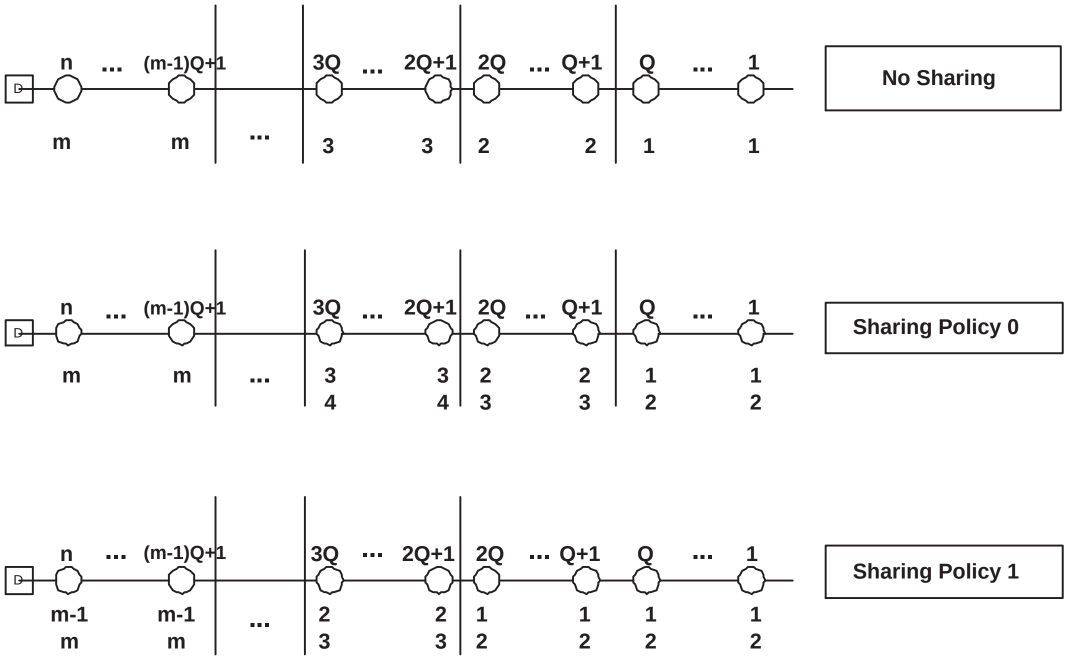

Figure 19: Comparison of different sharing policies

has additional routes 2 and m, $Cost(Sharing_1) \leq Cost(Sharing_0)$. So suppose route 1 is needed in policy 0 for a realization with $0 < X_1 < Q$. Let $\{n_1, n_2, ...n_m\}$ be the number of customers who place an order in that realization. Since route 1 is needed in policy 0, we must have $n_1 + n_2 > Q$ because otherwise route 1 would not be needed. In addition, we must have $n_1 + n_2 + n_3 > 2Q$ because otherwise $n_3 + (n_1 + n_2 - Q) \leq Q$ and route 1 would not be needed. In a similar manner, the fact that route 1 is needed for feasibility in policy 0 requires $\sum_{i=1}^{s} n_i > (s-1)Q$ for $s = 2, 3, ..., m$. But for these realizations, policy 1 results in the optimal solution with total cost $2(d_1 + d_{Q+1} + d_{2Q+1} + ... + d_{(m-1)Q+1})$, therefore we have $Cost(Sharing_1) \leq Cost(Sharing_0)$.

If $X_1 = Q$, furthest Q customers can be served by route 1 in policy 0 in the optimal solution, since route 1 cannot be used for other customers. Similarly furthest Q customers can be served by route 1 in policy 1 in the optimal solution too. So, for such realizations, we have the remaining routes to use in both policies as shown in Figure 21. Note that policy 1 has extra route m, but policy 0 has extra route 2. In realizations where route 2 is not needed in policy 0, route m is not needed for policy 1 either, so $Cost(Sharing_1) \leq$

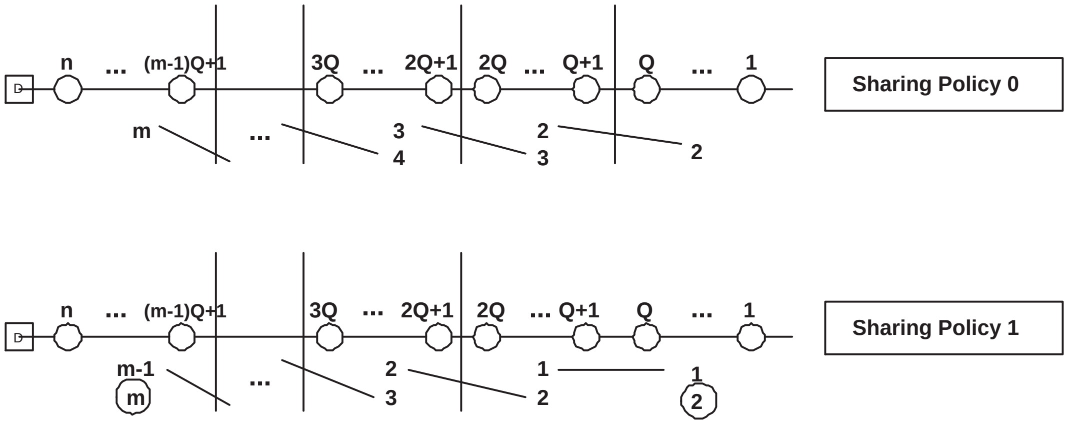

Figure 20: A dominance relation between two sharing policies if $0 < X_1 < Q$

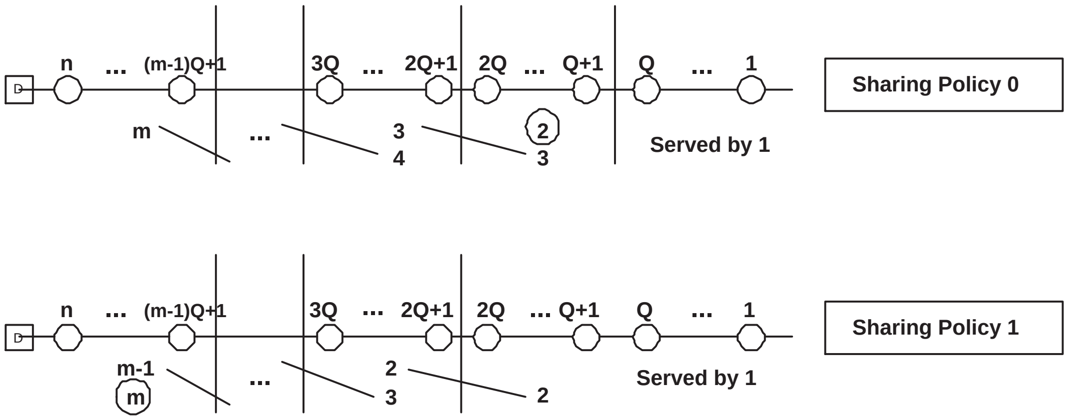

Figure 21: A dominance relation between two sharing policies if $X_1 = Q$, part a

$Cost(Sharing_0)$ holds. In realizations where route 2 is needed in policy 0, then we must have $\sum_{i=2}^{s} n_i > (s-2)Q$ for $s = 3, 4, ..., m$. Since route 2 is needed in policy 0, policy 0 serves all n_2 customers by route 2. Consequently, we have the following partial solution for both policies given in Figure 22. Note that since $\sum_{i=2}^{m} n_i > (m-2)Q$, we need at least $m-1$ vehicles to serve customers $n_2 + n_3 + ... + n_m$ and Sharing Policy 1 results in the lowest possible total cost because each route starting with 2 except m serves exactly Q customers. Therefore, $Cost(Sharing_1) \leq Cost(Sharing_0)$ holds.

Note that there are customer realizations where operational cost is strictly lower in policy 1 than policy 0, therefore we conclude Proposition 5.

□

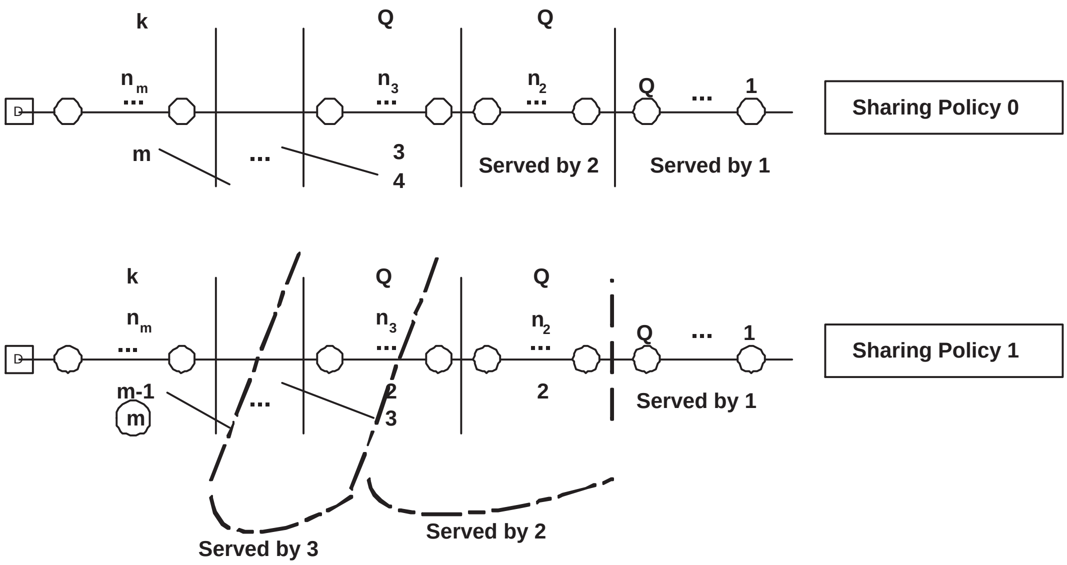

Figure 22: A dominance relation between two sharing policies if $X_1 = Q$, part b

It is likely to wonder whether sharing policy 1 is optimal among all possible planning level solutions. We next show that it is not optimal by finding another planning level solution with a smaller expected cost.

Observation 7 *There are policies which are better than sharing policy 1 in terms of expected total cost.*

We create a different policy and problem instance where sharing policy 1 given in Figure 19 gives a larger expected total cost and hence is not optimal. Consider the following two solutions presented in Figure 23. In solution 2, vehicle 2 has also been assigned $2Q$ customers. Among these $2Q$ customers, the furthest Q customers have backup vehicle 3, and closest Q customers have backup vehicle 4. So, in solution 2, each customer in the same fixed route does not necessarily have the same backup vehicle.

Note that solution 2 yields a lower operational cost under some demand realizations. For example if a single customer is arriving among furthest $2Q$ customers and a single customer is arriving among closest Q customers, then Solution 2 combines two customers into a single route (i.e. route 2), whereas Solution 1 needs two routes. On the other hand, there are some realizations where solution 1 yields a lower operational cost. For

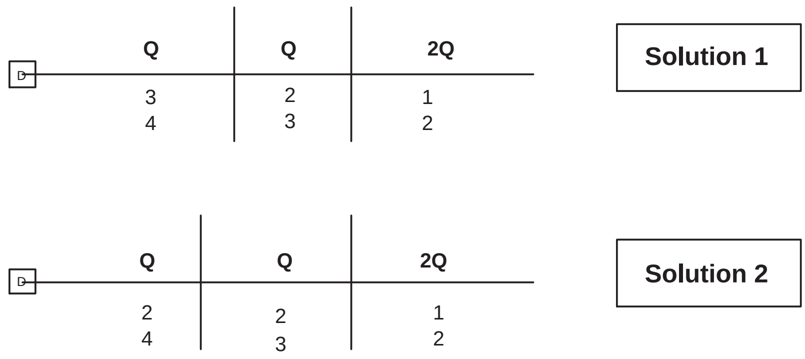

Figure 23: A different backup route scheme

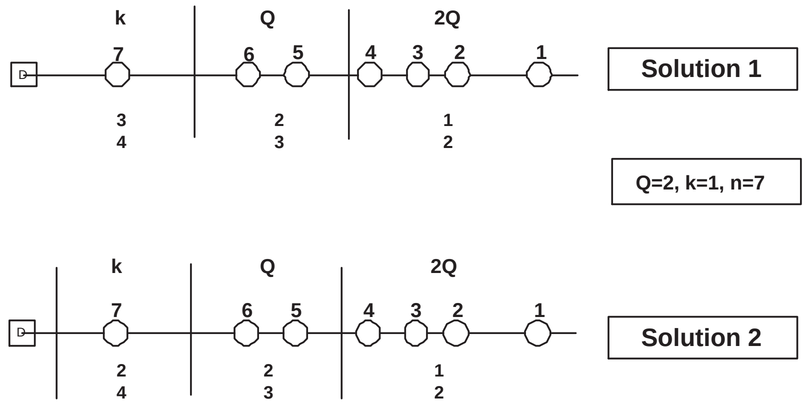

Figure 24: Comparison of two solutions belonging to two different backup schemes

example, consider the realization where all furthest $2Q$ customers, one customer among the closest Q customers, and one customer among the second closest Q customers are arriving. Then solution 2 would require 4 routes, whereas solution 1 would require only 3 routes. Therefore, we need to compare expected total costs of these two solutions over all customer realizations. We do this next on a simple problem instance given in Figure 24. We write these involved expressions in Appendix for this relatively small instance. If customers numbered as $\{1, 2, 3, 4, 5, 6, 7\}$ are located at distances $\{7, 6, 5, 4, 3, 2, 1\}$ and $p_i = 0.2$ for all customers, these expressions evaluate to:

E[Solution 1] = 4.241 and E[Solution 2] = 4.103, so that Solution 2 is better than Solution 1 in terms of expected total cost.

Observation 8 *For instances with $n > 3Q$, a two label scheme will not always allow the*

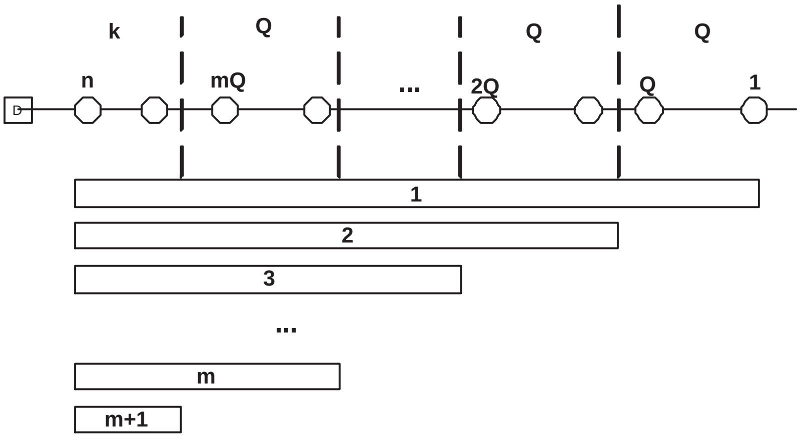

Figure 25: The minimum total labels to find the re-optimized solution in every customer realization

optimal operational solution.

A two label scheme with n customers assigns a total of $2n$ labels. However, if $n = mQ+k$ with $m \in Z^+$ and $n > 3Q$, then minimum total number of labels required to allow the optimal (re-optimized) operational solution for any customer realization is given by:

$T = n+(n-Q)+(n-2Q)+...+(n-mQ)$ as shown in Figure 25. All customers should have a common label (which we denote by label 1) because there is a realization where we have to combine any pair of customers together to find the re-optimized solution. For realizations where customers in the set $\{1, 2, .., Q\}$ are all arriving, all the other customers $\{Q + 1, Q + 2, ..., n\}$ should have a common distinct label, which we call label 2, and so on. But since $2n < n + (n - Q) + (n - 2Q) < T$, any two label scheme will have less than the required number of labels, hence yielding Observation 8.

Observation 9 *Sharing policy 1 is not optimal even among the class of solutions described by $x_i \geq x_j$, and $y_i = x_i + 1$ for any $i > j$*

Figure 26 shows two sharing policies in the class described by $x_i \geq x_j$, and $y_i = x_i+1$ for any $i > j$, where the boundaries are different. We have randomly created problem instances with $p_i = p$ for all i for the simple setting considered before, i.e. $Q = 2$, $k = 1$, and $n = 7$, and calculated expected costs of sharing policies 1 and 2 at different order probabilities.

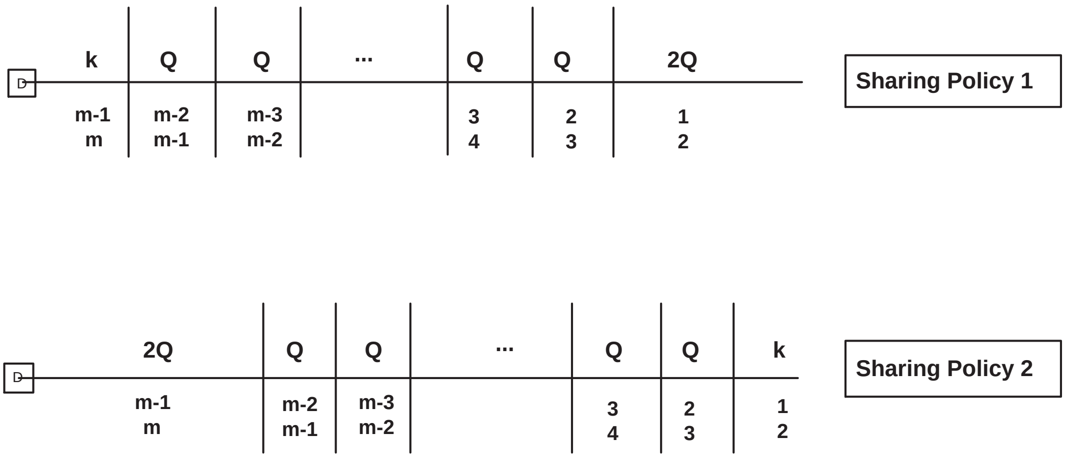

Figure 26: Comparison of two sharing policies of the same class

Figure 27 depicts the difference in expected costs at various order probabilities under 20 randomly created problem instances. As it can be seen, sharing policy 1 seems to outperform sharing policy 0 empirically after about $p > 0.5$ and is generally worse for $p < 0.5$. As p increases, sharing policy 1 seems to do much better empirically. This is intuitive because sharing policy 1 can combine customers into single vehicle further away from depot and when p increases, this opportunity arises more often. Note that because of the instances where $E[Sharing_2] - E[Sharing_1] < 0$, we conclude that sharing policy 1 is not optimal among all the solutions in this specific class.

3.6 Directions for Future Research

We have studied the VRPSCD in the context of stochastic unit demand customers that are located on the interval in this chapter. We characterized the optimum *a priori* routes in terms of expected total cost for problem instances where customers have order probabilities that are monotone non-decreasing from a depot located at the origin. For the case where customer order probabilities are not non-decreasing from depot, we show that optimal fixed routes can be found by solving a weighted matching problem for $Q = 2$. For arbitrary customer order probabilities, although we can introduce a pseudo-polynomial algorithm to find the optimal solution, it would be desirable to investigate whether finding optimal *a priori* routes for this case is NP-hard.

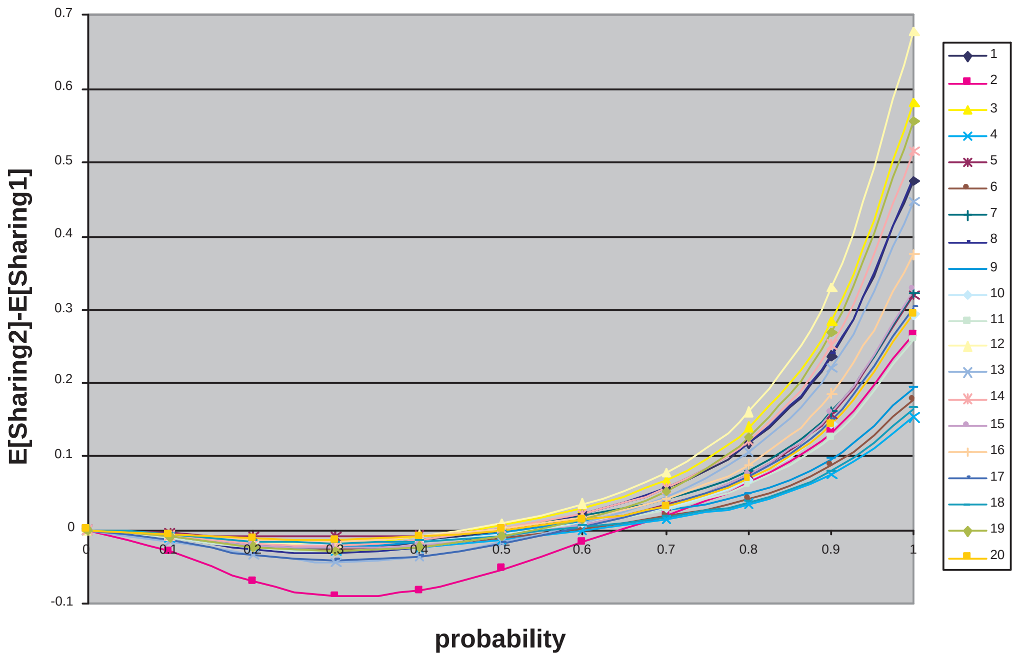

Figure 27: The difference in expected costs of two sharing policies vs probabilities

There exist many interesting future research areas for the Fixed Routes with Backup Vehicles Problem too. We have showed in this chapter that for a given planning level solution, i.e. (x_i, y_i) pair for each i, feasibility check can be performed by solving a maximum cardinality matching problem, but proved that operational (recourse) problem is NP-complete for a general feasible planning solution. Therefore, it is hard to find the optimal way of using fixed routes and backup vehicles even when customers are located on the interval so that difficulties associated with vehicle routing is removed. Fortunately, we could show that for a specific class of planning level solutions, feasibility check can be performed much faster and operational problem can be solved optimally in polynomial time. Furthermore, among the class of solutions whose recourse problem is easy to solve, we showed that the solution given by the boundaries $[k, Q, Q, ..., Q, Q, 2Q]$ is not always optimal for $p_i = p$ case because we could find a different solution which yields lower expected cost for low order probabilities. We were able to show these results by assuming that $p_i = p$, which simplified the objective function expressions to some extent using Binomial distribution on small problem instances. However, for larger problem instances and/or for different customer order probabilities, finding ways to evaluate expected cost expressions is a challenging task.

Throughout the chapter we assumed a conservative approach. Although it is straightforward to extend the results for traditional fixed routes if we relax this assumption, it is not trivial to do the same when the recourse strategy based on limited vehicle sharing is allowed.

And finally, no algorithm for finding fixed and backup route pairs has been proposed in this chapter, partially due to the fact that the recourse problem is difficult. However, it would be interesting to incorporate the recourse problem into *a priori* optimization stage by accepting sub-optimal solutions for the recourse problem. How to achieve this remains as a challenging opportunity.

CHAPTER IV

A DYNAMIC STOCHASTIC ROUTING PROBLEM

4.1 Introduction

We study a stochastic and dynamic uncapacitated routing problem that arises when there are certain service level agreements between customers and distributors in this chapter. In the specific setting we study, the distributor has to serve customer order within the next two days after the order is received but has the flexibility to choose the exact delivery date. Information about customer orders are revealed dynamically through time. Therefore, the distributor does not know future customer orders exactly but has probabilistic information regarding these orders. The objective is to minimize total delivery costs over the planning horizon by deciding when to serve each customer. We develop heuristic and optimal algorithms that use the probabilistic information about future orders for simple settings of this problem. Specifically, we consider single customer per period case where customers are located on the interval, on the circle and on the disk. We compare the performances of the suggested algorithms with those of dynamic algorithms that do not use future information and with off-line optimal solution which has perfect information about future orders. For general settings of the problem in which multiple customers that are distributed on the Euclidean plane may request delivery in a day, we modify heuristic algorithms suggested for simpler settings and perform a computational study to show the value of future information.

The classical Vehicle Routing Problem (VRP) that has been studied extensively in literature is both static and deterministic. However, in many practical applications, there are both stochastic and dynamic components of a problem. For example, customer demands (or at least all of them) are not usually known before the actual vehicle routes are constructed, but instead are revealed through time. Such problems are known as Dynamic Routing Problems (DRP) in which routing is a continuous process rather than static. With the help of recent advances in communication and information technologies, using dynamic routing methodologies in practice is becoming more applicable.

Stochastic and Dynamic Routing Problems (SDRP) arise when information about future customers to be visited are not known exactly at the time of decision making but instead are revealed dynamically, but the distribution planner has some probabilistic knowledge about the future customer orders. Most research in dynamic vehicle routing ignores stochastic information. There are only a few papers that tackle SDRPs with different objectives. Please refer to literature given in Section 1.5.

The specific routing problem we study has both stochastic and dynamic characteristics. The problem has been defined previously by Angelelli et al. [2]. It arises when there are certain Service Level Agreements (SLAs) between the end customer and the distributor. Specifically, the distributor has to deliver a customer order anytime within the next two business days after it has been received. So, at the beginning of a day, there are two sets of customers. In the first set, there are customers that must served today because their service has already been delayed the previous day. In the second set, there are newly arriving customers who may either be served today or who may be delayed to the following day. The planner does not know customer orders further into future exactly, but knows them in distribution. The objective is to minimize total transportation costs over a planning horizon. We analyze this problem from both an academic and practical point of view. We study the problem settings where customers are distributed on the interval, on the circle, on the disk and finally on the the Euclidean plane and develop exact and heuristic algorithms for different problem settings.

The chapter is organized as follows: Section 4.2 formally defines the problem setting in which customers are distributed on the interval with a known distribution function. In Sections 4.3 and 4.4, we extend the results to the settings where customers are distributed on the circle and on the disk respectively. Section 4.5 considers the most general settings of the problem and presents heuristic implementations of algorithms suggested earlier. In each section, we first present suggested algorithms and then conduct a computational study to compare the efficacy of the suggested approaches. Finally, Section 4.6 introduces potential research directions.

4.2 DSRP on the Interval

The Dynamic Stochastic Routing Problem on the Interval (DSRP-I) is defined as follows. At the beginning of each period $t = 0, 1, \ldots$ a request is received to visit a customer located at point $\xi_t \in [0, 1]$. A customer who places a request at the beginning of period t has to be visited in period t or in period $t + 1$. In each period t, a vehicle starts at point 0, travels to the customer(s) that are visited in period t, if any, and then returns to point 0. We sometimes refer to point 0 as the depot. The cost incurred in period t is equal to the distance traveled in period t, which is equal to twice the distance to the furthest point visited. For example, suppose that the customer whose request was received in period $t-1$ was visited in period $t-1$. In period t, one may decide to visit the customer whose request is received in period t, in which case the cost incurred in period t is equal to $2\xi_t$. One may also decide not to visit the customer, in which case no cost is incurred in period t. Next, suppose that the customer whose request was received in period $t - 1$ was not visited in period $t - 1$. That customer has to be visited in period t. In period t, one may decide not to visit the customer whose request was received in period t, in which case the cost incurred in period t is equal to $2\xi_{t-1}$. One may also decide to visit the customer whose request was received in period t, in which case the cost incurred in period t is equal to $2 \max\{\xi_{t-1}, \xi_t\}$.

Let $u_t \in \{0, 1\}$ denote the decision at time t, where $u_t = 0$ denotes that the customer whose request is received at time t is not visited at time t, and $u_t = 1$ denotes that the customer whose request is received at time t is visited at time t. Let

$$f(\xi, u) \quad := \quad \xi(1 - u).$$

Then $x_t = f(\xi_{t-1}, u_{t-1})$ denotes the location of the customer whose request was received at time $t - 1$ and who has to be visited at time t; that is, if the customer whose request was received at time $t-1$ was visited at time $t-1$, then $x_t = 0$, and if the customer whose request was received at time $t-1$ was not visited at time $t-1$, then $x_t = \xi_{t-1}$. Let

$$c(x, \xi, u) \quad := \quad \max\{x, \xi u\}.$$

Then the cost incurred at time t is given by $2c(x_t, \xi_t, u_t)$. For ease of presentation, we will use $c(x_t, \xi_t, u_t)$ as the cost in the remainder of this section. As this simply scales down the cost, it will not affect the decisions and thus the dispatch policies in any way.

We assume that x_0 is given, and that $\{\xi_t\}_{t=0}^{\infty}$ is an independent and identically distributed sequence with common distribution function F.

4.2.1 Infinite Horizon Optimal Policy

In this section, we characterize the optimal policy for the DSRP-I, and discuss computation of the optimal policy. Let $\alpha \in (0,1)$ denote the discount factor per time period. The objective is to minimize the expected total discounted cost over the planning horizon.

Let Π denote the set of all measurable functions $\pi : [0,1] \times [0,1] \mapsto \{0,1\}$ representing the stationary deterministic policies. Then the infinite horizon problem is

$$\min_{\pi \in \Pi} \mathbb{E}\left[\sum_{t=0}^{\infty} \alpha^t c(x_t, \xi_t, \pi(x_t, \xi_t))\right]. \tag{4}$$

Let the optimal value function $V^* : [0,1] \mapsto \mathbb{R}$ be given by

$$V^*(x) \quad := \quad \min_{\pi \in \Pi} \mathbb{E}\left[\sum_{t=0}^{\infty} \alpha^t c(x_t, \xi_t, \pi(x_t, \xi_t)) \middle| x_0 = x\right]. \tag{5}$$

4.2.1.1 Properties of the Optimal Value Function and Optimal Policy

The optimal value function V^* satisfies the following optimality equation:

$$\begin{aligned} V^*(x) &= \mathbb{E}_F\left[\min_{u \in \{0,1\}} \{c(x,\xi,u) + \alpha V^*(f(\xi,u))\}\right] \\ &= \int_{[0,1]} \min\{x + \alpha V^*(\xi),\ \max\{x,\xi\} + \alpha V^*(0)\}\ dF(\xi). \end{aligned} \tag{6}$$

It follows by observing the right side of the optimality equation (6) that the optimal value function V^* is nondecreasing. Proposition 6 establishes that the optimal value function V^* is Lipschitz continuous with Lipschitz constant 1.

Proposition 6 *For any $0 \le x_1 \le x_2 \le 1$ it holds that*

$$0 \quad \le \quad V^*(x_2) - V^*(x_1) \quad \le \quad x_2 - x_1.$$

When we address computational issues, we are also interested in functions $V : [0,1] \mapsto \mathbb{R}$ that approximate V^*. We will show that it is sufficient to restrict attention to functions V that satisfy $0 \leq V(x_2) - V(x_1) \leq x_2 - x_1$ for all $0 \leq x_1 \leq x_2 \leq 1$.

An optimal policy can be characterized in various ways. Given the location ξ of the new arrival, one may characterize the set

$$X^*(\xi) \quad := \quad \{x \in [0,1] \ : \ x + \alpha V^*(\xi) \geq \max\{x, \xi\} + \alpha V^*(0)\}$$

of locations of the customer remaining from the previous time period such that the new customer should be visited immediately. In general, for any function $V : [0,1] \mapsto \mathbb{R}$, let

$$X(\xi) \quad := \quad \{x \in [0,1] \ : \ x + \alpha V(\xi) \geq \max\{x, \xi\} + \alpha V(0)\}.$$

Note that, if V is nondecreasing, then for any $x \geq \xi$ it holds that $x + \alpha V(\xi) = \max\{x, \xi\} + \alpha V(\xi) \geq \max\{x, \xi\} + \alpha V(0)$, and thus $X(\xi) \supset [\xi, 1]$. Hence, it follows from V^* being nondecreasing and the optimality equation (6) that if the customer whose request is received at time t is located closer to the depot than the customer remaining from time $t-1$, i.e., $\xi_t \leq x_t$, then it is optimal to immediately visit the customer whose request is received at time t. For any function $V : [0,1] \mapsto \mathbb{R}$, let $\zeta : [0,1] \mapsto [0,1]$ be given by

$$\zeta(\xi) \quad := \quad \xi + \alpha V(0) - \alpha V(\xi).$$

Specifically, let

$$\zeta^*(\xi) \quad := \quad \xi + \alpha V^*(0) - \alpha V^*(\xi).$$

If $V(x_2) - V(x_1) \leq x_2 - x_1$ for all $0 \leq x_1 \leq x_2 \leq 1$, then $\alpha[V(\xi_2) - V(\xi_1)] \leq \alpha[\xi_2 - \xi_1] \leq \xi_2 - \xi_1$, and thus $\zeta(\xi_1) \leq \zeta(\xi_2)$, that is, ζ is nondecreasing. Specifically, $\zeta(\xi) \geq \zeta(0) = 0$. Also, if V is nondecreasing then $\zeta(\xi_2) - \zeta(\xi_1) \leq \xi_2 - \xi_1$, and thus ζ is Lipschitz continuous with Lipschitz constant 1. Specifically, $\zeta(\xi) \leq \xi \leq 1$.

Consider any $\xi \in [0,1]$ and any $x < \zeta(\xi)$. Then $x + \alpha V(\xi) < \xi + \alpha V(0) = \max\{x, \xi\} + \alpha V(0)$. Next, consider any $\xi \in [0,1]$ and any $x \geq \zeta(\xi)$. If $x \leq \xi$, then $x + \alpha V(\xi) \geq \xi + \alpha V(0) = \max\{x, \xi\} + \alpha V(0)$. If $x \geq \xi$, then it follows as remarked before that $x + \alpha V(\xi) \geq \max\{x, \xi\} + \alpha V(0)$. It follows that $X(\xi) = [\zeta(\xi), 1]$. Specifically, $X^*(\xi) = [\zeta^*(\xi), 1]$, and hence an optimal policy is given by the threshold function ζ^*. Thus, if the customer remaining from the previous time period is located at x and the new arrival is located at ξ and $x \geq \zeta^*(\xi)$, then the new arrival should be visited immediately. Otherwise, if $x < \zeta^*(\xi)$, then service of the new arrival should be delayed.

It may be more natural to characterize an optimal policy in the following way. For any function $V : [0,1] \mapsto \mathbb{R}$, and any location x of the customer remaining from the previous time period, let

$$\Xi(x) \quad := \quad \{\xi \in [0,1] \; : \; x + \alpha V(\xi) \geq \max\{x, \xi\} + \alpha V(0)\}.$$

Specifically, let

$$\Xi^*(x) \quad := \quad \{\xi \in [0,1] \; : \; x + \alpha V^*(\xi) \geq \max\{x, \xi\} + \alpha V^*(0)\}$$

denote the set of locations of the newly arriving customer such that the new customer should be visited immediately. Recall that if V is nondecreasing, then for any $\xi \leq x$ it holds that $x + \alpha V(\xi) = \max\{x, \xi\} + \alpha V(\xi) \geq \max\{x, \xi\} + \alpha V(0)$, and thus $\Xi(x) \supset [0, x]$. Consider the function $z : [0,1] \mapsto [0,1]$ given by

$$z(x) \quad := \quad \sup \, \Xi(x)$$

and specifically, let

$$z^*(x) \quad := \quad \sup \, \Xi^*(x).$$

If V is nondecreasing, then $z(x) \geq x$. Other properties of z are explored later. Next we show that if $0 \leq V(x_2) - V(x_1) \leq x_2 - x_1$ for all $0 \leq x_1 \leq x_2 \leq 1$, then $\Xi(x) = [0, z(x)]$. It follows that it is optimal to visit a newly arriving customer located at ξ if $\xi \leq z^*(x)$, and it is optimal not to visit the newly arriving customer if $\xi > z^*(x)$.

Consider $\xi_1 < \xi_2$, and suppose that $x + \alpha V(\xi_1) < \max\{x, \xi_1\} + \alpha V(0)$, that is, $\xi_1 \notin \Xi(x)$. Note that it follows that $x < \xi_1$. We want to show that $\xi_2 \notin \Xi(x)$, that is, $x + \alpha V(\xi_2) < \max\{x, \xi_2\} + \alpha V(0)$. Note that it would be sufficient to show that the increase $[x + \alpha V(\xi_2)] - [x + \alpha V(\xi_1)]$ is less than the increase $[\max\{x, \xi_2\} + \alpha V(0)] - [\max\{x, \xi_1\} + \alpha V(0)]$. We establish that next:

$$
\begin{aligned}
x+\alpha V(\xi_2) &= [x+\alpha V(\xi_2)]-[x+\alpha V(\xi_1)]+[x+\alpha V(\xi_1)] \\
&< \alpha[V(\xi_2)-V(\xi_1)]+\max\{x,\xi_1\}+\alpha V(0) \\
&\le \alpha[\xi_2-\xi_1]+\xi_1+\alpha V(0) \\
&\le \xi_2+\alpha V(0) \quad = \quad \max\{x,\xi_2\}+\alpha V(0).
\end{aligned}
$$

The second inequality follows from the assumption that $V(x_2)-V(x_1)\le x_2-x_1$ for all $0\le x_1\le x_2\le 1$, and the observation that V nondecreasing and $x+\alpha V(\xi_1)<\max\{x,\xi_1\}+\alpha V(0)$ imply that $x<\xi_1$. It follows that $x+\alpha V(\xi)\ge\max\{x,\xi\}+\alpha V(0)$ for all $\xi\in[0,z(x))$, that is, $\Xi(x)\supset[0,z(x))$. Also, it follows from the continuity of V that $x+\alpha V(z(x))\ge\max\{x,z(x)\}+\alpha V(0)$ (and if $z(x)<1$, then it follows from the intermediate value theorem that $x+\alpha V(z(x))=\max\{x,z(x)\}+\alpha V(0)$), and thus $\Xi(x)=[0,z(x)]$. Hence, an optimal policy is given by the threshold function z^*. In particular, if the customer remaining from the previous time period is located at x and the newly arriving customer is located at $\xi\le z^*(x)$, then it is optimal to visit the newly arriving customer immediately, otherwise it is optimal to delay the visit of the newly arriving customer.

It follows that the optimality equation (6) can be written as follows:

$$
\begin{aligned}
V^*(x) &= \int_{[0,z^*(x)]}[\max\{x,\xi\}+\alpha V^*(0)]\ dF(\xi)+\int_{(z^*(x),1]}[x+\alpha V^*(\xi)]\ dF(\xi) \\
&= \int_{[0,x]}[x+\alpha V^*(0)]\,dF(\xi)+\int_{(x,z^*(x)]}[\xi+\alpha V^*(0)]\,dF(\xi) \\
&+\int_{(z^*(x),1]}[x+\alpha V^*(\xi)]\,dF(\xi).
\end{aligned}
\tag{7}
$$

Recall that if V is nondecreasing, then $z(x)\ge x$. It follows immediately that $z(1)=1$. Next, we show that $z(0)=0$. Thus, if no customer remains from the previous time period (or a customer located at 0), then it is optimal to delay the visit of any newly arriving customer. Consider $x = 0$ and any $\xi > 0$. Then

$$\begin{aligned} x + \alpha V(\xi) &= \alpha[V(\xi) - V(x)] + x + \alpha V(x) \\ &\leq \alpha[\xi - x] + x + \alpha V(x) \\ &< \xi + \alpha V(0) \\ &= \max\{x, \xi\} + \alpha V(0). \end{aligned}$$

The first inequality follows from the assumption that $V(x_2) - V(x_1) \leq x_2 - x_1$ for all $0 \leq x_1 \leq x_2 \leq 1$. It follows that $z(0) = 0$.

Next we establish additional properties of z, such as monotonicity and bounds on the slope of z, and thus continuity of z. First consider the following example. Consider functions $f_1, f_2 : [0,1] \times [0,1] \mapsto \mathbb{R}$. For each $x \in [0,1]$, let

$$z(x) \quad := \quad \sup\{\xi \in [0,1] \ : \ f_1(x,\xi) \geq f_2(x,\xi)\}.$$

It does not hold that continuity of f_1 and f_2 implies continuity of z. For example, let

$$\begin{aligned} f_1(x,\xi) &= 1/2 \\ f_2(x,\xi) &:= \begin{cases} 2\xi & \text{if} \quad \xi \leq x/2 \\ x & \text{if} \quad x/2 \leq \xi \leq x/2 + 1/2 \\ -1 + 2\xi & \text{if} \quad x/2 + 1/2 \leq \xi \end{cases} \end{aligned}$$

Then

$$z(x) = \begin{cases} 3/4 & \text{if} \quad x \leq 1/2 \\ 1/4 & \text{if} \quad x > 1/2 \end{cases}$$

Consider $0 \leq x_1 \leq x_2 \leq 1$, and any $\xi \in [0, z(x_1)]$. Note that

$$\max\{x_2, \xi\} - \max\{x_1, \xi\} \quad \leq \quad x_2 - x_1.$$

Thus

$$[x_2 + \alpha V(\xi)] - [\max\{x_2, \xi\} + \alpha V(0)] \quad \geq \quad [x_1 + \alpha V(\xi)] - [\max\{x_1, \xi\} + \alpha V(0)] \quad \geq \quad 0.$$

It follows from the definition of z that $z(x_2) \geq \xi$ for all $\xi \in [0, z(x_1)]$, and thus $z(x_2) \geq z(x_1)$, that is, z is a nondecreasing function.

Consider any $0 \leq x_1 \leq x_2 \leq 1$ and any $\xi \in [x_1, z(x_1)]$ such that $\xi + x_2 - x_1 \leq 1$. Then

$$
\begin{aligned}
& [x_2 + \alpha V(\xi + x_2 - x_1)] - [\max\{x_2, \xi + x_2 - x_1\} + \alpha V(0)] \\
& \geq [x_2 + \alpha V(\xi)] - [\xi + x_2 - x_1 + \alpha V(0)] \\
& = [x_1 + \alpha V(\xi)] - [\xi + \alpha V(0)] \\
& = [x_1 + \alpha V(\xi)] - [\max\{x_1, \xi\} + \alpha V(0)] \\
& \geq 0.
\end{aligned}
$$

It follows from the definition of z that $z(x_2) \geq \xi + x_2 - x_1$ for all $\xi \in [x_1, z(x_1)]$, and thus $z(x_2) \geq z(x_1) + x_2 - x_1$, or $z(x_2) - z(x_1) \geq x_2 - x_1$, that is, z increases at rate at least 1 until $z(x) = 1$, whereafter z remains constant.

Next we establish an upper bound on the rate of increase of z. Consider any $0 \leq x_1 \leq x_2 \leq 1$. If $z(x_1) = 1$, then $z(x_2) = 1$, and thus $z(x_2) - z(x_1) = 0$. Next suppose that $z(x_1) < 1$. Consider any $\xi_1 \in (z(x_1), 1]$, and any $\xi_2 \geq \xi_1 + (x_2 - x_1)/(1-\alpha)$. If $\xi_2 > 1$, then $z(x_2) \leq \xi_1 + (x_2 - x_1)/(1-\alpha)$. Otherwise, note that $x_1 + \alpha V(\xi_1) < \max\{x_1, \xi_1\} + \alpha V(0) = \xi_1 + \alpha V(0)$. Recall that $\alpha[V(\xi_2) - V(\xi_1)] \leq \alpha[\xi_2 - \xi_1]$. Then it follows that

$$
\begin{array}{rrcl}
 & [\xi_2 - x_2 - \alpha V(\xi_2)] - [\xi_1 - x_1 - \alpha V(\xi_1)] & \geq & [\xi_2 - \xi_1] - (x_2 - x_1) - \alpha[\xi_2 - \xi_1] \quad \geq \quad 0 \\
\Rightarrow & [\max\{x_2, \xi_2\} + \alpha V(0)] - [x_2 + \alpha V(\xi_2)] & \geq & [\xi_2 + \alpha V(0)] - [x_2 + \alpha V(\xi_2)] \\
 & & \geq & [\xi_1 + \alpha V(0)] - [x_1 + \alpha V(\xi_1)] \quad > \quad 0
\end{array}
$$

It follows from the definition of z that for any $\xi_1 > z(x_1)$, $z(x_2) \leq \xi_1 + (x_2 - x_1)/(1-\alpha)$, and thus $z(x_2) \leq z(x_1) + (x_2 - x_1)/(1-\alpha)$, hence $0 \leq z(x_2) - z(x_1) \leq (x_2 - x_1)/(1-\alpha)$. Thus z is Lipschitz continuous with Lipschitz constant $1/(1-\alpha)$. In particular, it follows from V^* being nondecreasing and Proposition 6 that the results above hold for $V = V^*$ and $z = z^*$.

4.2.1.2 Computation of the Optimal Value Function and Optimal Policy

The optimality equation (6) can be used to compute (approximately) the optimal value function V^* and thereby an (approximately) optimal policy. However, there are two obstacles to be overcome:

1. The unknown optimal value function V^* also appears on the right side of the optimality equation (6). This is a standard problem in dynamic programming and we will address that in the standard way.

2. One may not be able to compute the integral on the right side of the optimality equation (6) exactly. We will consider a method to approximate the integral, and we will derive error bounds.

First, we consider a mapping representing the right side of the optimality equation. Let $\mathcal{V}$ denote the set of bounded functions $V : [0,1] \mapsto \mathbb{R}$, and let $\|V\| := \sup_{x\in[0,1]} |V(x)|$. Consider the mapping $T^* : \mathcal{V} \mapsto \mathcal{V}$ given by

$$T^*(V)(x) \quad := \quad \int_{[0,1]} \min\left\{x + \alpha V(\xi),\ \max\{x,\xi\} + \alpha V(0)\right\}\ dF(\xi)$$

Note that T^* has contraction factor α and unique fixed point V^*. We will approximate the integral by replacing F with an approximating distribution $\hat{F}$. Then the corresponding mapping $\hat{T} : \mathcal{V} \mapsto \mathcal{V}$ is given by

$$\hat{T}(V)(x) \quad := \quad \int_{[0,1]} \min\left\{x + \alpha V(\xi),\ \max\{x,\xi\} + \alpha V(0)\right\}\ d\hat{F}(\xi)$$

and has contraction factor α and unique fixed point $\hat{V}$. We will choose an error tolerance $\varepsilon > 0$, and then we will choose $\hat{F}$ such that for any $V \in \mathcal{V}$, $\hat{T}(V)$ is easy to compute, and for any appropriate (Lipschitz continuous with Lipschitz constant 1) $V \in \mathcal{V}$ and any $x \in [0,1]$,

$$\left|T^*(V)(x) - \hat{T}(V)(x)\right| \quad \leq \quad \varepsilon$$

that is, $\left\|T^*(V) - \hat{T}(V)\right\| \leq \varepsilon$. Next we show that as a result the fixed points V^* and $\hat{V}$ will be close to each other.

Lemma 7 *Consider two mappings $T_1, T_2 : \mathcal{V} \mapsto \mathcal{V}$ with fixed points V_1, V_2 respectively. Suppose that $\|T_2(V_1) - T_1(V_1)\| \leq \varepsilon$, and that T_2 has contraction factor $\alpha < 1$. Then*

$$\|V_2 - V_1\| \quad \leq \quad \frac{\varepsilon}{1-\alpha}$$

In addition to approximating T^* with $\hat{T}$, we will approximate $\hat{V}$ with $\hat{T}^i(V_0)$ for some initial function V_0 and sufficiently large i. Note that for any $V_0 \in \mathcal{V}$, $\left\|\hat{T}^i(V_0) - \hat{T}^{i-1}(V_0)\right\| = \left\|\hat{T}^{i-1}(\hat{T}(V_0)) - \hat{T}^{i-1}(V_0)\right\| \leq \alpha^{i-1}\left\|\hat{T}(V_0) - V_0\right\| \to 0$ as $i \to \infty$. Next we show that if calculations stop when $\left\|\hat{T}(V) - V\right\|$ is small for some $V \in \mathcal{V}$ (such as $V = \hat{T}^{i-1}(V_0)$), then $\hat{T}(V)$ is close to $\hat{V}$ (and thus also close to V^*).

Lemma 8 *Consider a mapping $T : \mathcal{V} \mapsto \mathcal{V}$ with contraction factor $\alpha < 1$ and fixed point V^∞. Suppose that for some $V \in \mathcal{V}$,*

$$\|T(V) - V\| \quad \leq \quad \vartheta$$

Then

$$\|V^\infty - T(V)\| \quad \leq \quad \frac{\alpha\vartheta}{1-\alpha}$$

Proposition 9 summarizes the results so far regarding approximation of the optimal value function.

Proposition 9 *Suppose that*

$$\left\|T^*(V^*) - \hat{T}(V^*)\right\| \quad \leq \quad \varepsilon$$

and for some $V \in \mathcal{V}$,

$$\left\|\hat{T}(V) - V\right\| \quad \leq \quad \vartheta$$

Then

$$\left\|V^* - \hat{T}(V)\right\| \quad \leq \quad \frac{\varepsilon + \alpha\vartheta}{1-\alpha}$$

Suppose that calculations stop when $\left\|\hat{T}(V) - V\right\|$ is small for some $V \in \mathcal{V}$, and a policy based on $\hat{T}^2(V)$ is chosen as described next (for a reason to be explained later, the policy is based on $\hat{T}^2(V)$ and not $\hat{T}(V)$). Next we show that the resulting policy is almost optimal.

Proposition 10 *Suppose that*

$$\left\|T^*(V^*) - \hat{T}(V^*)\right\| \quad \leq \quad \varepsilon$$

and for some $V \in \mathcal{V}$,

$$\left\|T^*(\hat{T}^2(V)) - \hat{T}(\hat{T}^2(V))\right\| \leq \varepsilon$$

and

$$\left\|\hat{T}(V) - V\right\| \leq \vartheta$$

Consider the policy $\hat{\pi} : [0,1] \times [0,1] \mapsto \{0,1\}$ *given by*

$$\begin{aligned}
\hat{\pi}(x,\xi) &:= \arg\max_u \left\{c(x,\xi,u) + \alpha\hat{T}^2(V)(f(x,\xi,u))\right\} \\
&= \begin{cases} 1 & \text{if} \quad x + \alpha\hat{T}^2(V)(\xi) \geq \max\{x,\xi\} + \alpha\hat{T}^2(V)(0) \\ 0 & \text{otherwise} \end{cases}
\end{aligned}$$

Let the value function of policy $\hat{\pi}$ *be given by*

$$V^{\hat{\pi}}(x) := \mathbb{E}_{\{\xi_t\}}\left[\sum_{t=0}^{\infty} \alpha^t c(x_t,\xi_t,\hat{\pi}(x_t,\xi_t)) \middle| x_0 = x\right]$$

Then

$$\left\|V^* - V^{\hat{\pi}}\right\| \leq 2\frac{\varepsilon + \alpha^2\vartheta}{1-\alpha}$$

Next, we illustrate a way of choosing $\hat{F}$, and verify that it satisfies the properties above.

Approximation $\hat{F}$ by discretization of the support of F. Consider a continuous function $f : [0,1] \mapsto \mathbb{R}$, and a probability distribution F on $[0,1]$. Next we approximate $\int_{[0,1]} f(y)dF(y)$ by discretization of the support $[0,1]$ of F. First, consider any approximating function $\hat{f} : [0,1] \mapsto \mathbb{R}$ such that $\left|f(y) - \hat{f}(y)\right| \leq \varepsilon$ for all $y \in [0,1]$. Then

$$\left|\int_{[0,1]} f(y)dF(y) - \int_{[0,1]} \hat{f}(y)\, dF(y)\right| \leq \int_{[0,1]} \left|f(y) - \hat{f}(y)\right| dF(y) \leq \varepsilon$$

Because f is uniformly continuous on $[0,1]$, for any $\varepsilon > 0$ there is a $\delta > 0$ such that $|f(y_2) - f(y_1)| \leq \varepsilon$ for all $y_1, y_2 \in [0,1]$ with $|y_2 - y_1| \leq \delta$. For example, if f is Lipschitz continuous with Lipschitz constant L, then by choosing any $\delta \leq \varepsilon/L$ it follows that $|f(y_2) - f(y_1)| \leq L\,|y_2 - y_1| \leq L\delta \leq \varepsilon$. Next choose any positive integer $m \geq 1/\delta$, and let $y_{m,j} \in [(j-1)/m, j/m]$ for $j = 1,\ldots,m$. Define $f_m : [0,1] \mapsto \mathbb{R}$ by $f_m(0) := f(y_{m,1})$, and for any $y \in ((j-1)/m, j/m]$, $f_m(y) := f(y_{m,j})$. Note that $|f(0) - f_m(0)| = |f(0) - f(y_{m,1})| \leq$

ε, and for any $y \in ((j-1)/m, j/m]$, $|f(y) - f_m(y)| = |f(y) - f(y_{m,j})| \leq \varepsilon$ because $|y - y_{m,j}| \leq 1/m \leq \delta$. Thus $\left|\int_{[0,1]} f(y)dF(y) - \int_{[0,1]} f_m(y)dF(y)\right| \leq \varepsilon$.

Note that

$$\begin{aligned}
\int_{[0,1]} f_m(y)\, dF(y) &= f(y_{m,1})F(1/m) + \sum_{j=2}^{m} f(y_{m,j})[F(j/m) - F((j-1)/m)] \\
&= f(y_{m,1})F_m(y_{m,1}) + \sum_{j=2}^{m} f(y_{m,j})[F_m(y_{m,j}) - F_m(y_{m,j-1})] \\
&= \int_{[0,1]} f(y)\, dF_m(y)
\end{aligned}$$

where F_m is the discrete distribution function given by

$$F_m(y) \;:=\; F(1/m)\mathbb{I}_{\{y \geq y_{m,1}\}} + \sum_{j=2}^{m} [F(j/m) - F((j-1)/m)]\mathbb{I}_{\{y \geq y_{m,j}\}}$$

It remains to show that each integrand of interest is Lipschitz continuous. It follows from Lemma 18 in the appendix that for any function $V : [0,1] \mapsto \mathbb{R}$, any distribution function F, and any $0 \leq x_1 \leq x_2 \leq 1$, it holds that

$$0 \;\leq\; T(V)(x_2) - T(V)(x_1) \;\leq\; x_2 - x_1$$

Note that this also holds for $\hat{F}$ and $\hat{T}$. Thus, for any x, and any $0 \leq \xi_1 \leq \xi_2 \leq 1$, it holds that

$$\begin{aligned}
0 \;&\leq\; [x + \alpha T(V)(\xi_2)] - [x + \alpha T(V)(\xi_1)] \;\leq\; \alpha(\xi_2 - \xi_1) \\
0 \;&\leq\; [\max\{x, \xi_2\} + \alpha T(V)(0)] - [\max\{x, \xi_1\} + \alpha T(V)(0)] \;\leq\; \xi_2 - \xi_1
\end{aligned}$$

Hence it follows from Lemma 16 in the appendix that for each x, and any $0 \leq \xi_1 \leq \xi_2 \leq 1$,

$$\begin{aligned}
0 \;\leq\;& \min\{x + \alpha T(V)(\xi_2),\ \max\{x, \xi_2\} + \alpha T(V)(0)\} \\
& - \min\{x + \alpha T(V)(\xi_1),\ \max\{x, \xi_1\} + \alpha T(V)(0)\} \\
\leq\;& \xi_2 - \xi_1
\end{aligned}$$

Therefore, for any function $V : [0,1] \mapsto \mathbb{R}$ and any x, the integrand:

$\min\{x + \alpha T(V)(\xi),\ \max\{x, \xi\} + \alpha T(V)(0)\}$ is Lipschitz continuous with Lipschitz constant 1. Hence, for any function $V : [0,1] \mapsto \mathbb{R}$, any distribution function F, and for any

$\varepsilon > 0$, we can choose any integer $m \geq 1/\varepsilon$ and any points $y_{m,j} \in [(j-1)/m, j/m]$ for $j = 1, \ldots, m$, to obtain

$$\begin{aligned}
& \|T^*(T(V)) - T_m(T(V))\| \\
= & \left| \int_{[0,1]} \min\{x + \alpha T(V)(\xi),\ \max\{x, \xi\} + \alpha T(V)(0)\}\ dF(\xi) \right. \\
& - \Big[\min\{x + \alpha T(V)(y_{m,1}),\ \max\{x, y_{m,1}\} + \alpha T(V)(0)\} F(1/m) \\
& \left. + \sum_{j=2}^{m} \min\{x + \alpha T(V)(y_{m,j}),\ \max\{x, y_{m,j}\} + \alpha T(V)(0)\} [F(j/m) - F((j-1)/m)] \Big] \right| \\
\leq & \ \varepsilon
\end{aligned}$$

Since V^* is Lipschitz continuous with Lipschitz constant 1, it follows that $\|T^*(V^*) - T_m(V^*)\| \leq \varepsilon$. Therefore, the distribution function $\hat{F} = F_m$ and the resulting mapping $\hat{T} = T_m$ satisfy the assumptions of Propositions 9 and 10.

Computational considerations. The results above enable one to compute an ϵ-optimal policy, that is, a policy $\hat{\pi}$ with $\left\|V^* - V^{\hat{\pi}}\right\| \leq \epsilon$. For example, given $\epsilon > 0$, choose $\varepsilon = (1-\alpha)\epsilon/4$ and $\vartheta = (1-\alpha)\epsilon/(4\alpha^2)$. Then choose m as described above so that the resulting mapping T_m satisfies $\|T^*(V) - T_m(V)\| \leq \varepsilon$ for all Lipschitz continuous functions $V : [0,1] \mapsto \mathbb{R}$ with Lipschitz constant 1. Choose any initial function $V_0 : [0,1] \mapsto \mathbb{R}$. Then for $i = 1, 2, \ldots$, compute $T_m^i(V_0) = T_m(T_m^{i-1}(V_0))$. Since $\|T_m^i(V_0) - T_m^{i-1}(V_0)\| \leq \alpha^{i-1}\|T_m(V_0) - V_0\| \to 0$ as $i \to \infty$, it follows that for $i \geq [\log(\vartheta) - \log(\|T_m(V_0) - V_0\|)]/\log(\alpha) + 1$ it holds that $\|T_m^i(V_0) - T_m^{i-1}(V_0)\| \leq \vartheta$. (We discuss some issues regarding the calculation of $\|T_m^i(V_0) - T_m^{i-1}(V_0)\|$ soon.) At this point the algorithm stops. The resulting policy π_m^{i+1} is defined as follows. Consider any given $x, \xi \in [0,1]$. Note that it does not necessarily hold that $\xi \in \{0, y_{m,1}, \ldots, y_{m,m}\}$, and thus it may be the case that $T_m^i(V_0)(\xi)$ has not been computed (this point is discussed in more detail soon). Instead of computing $T_m^i(V_0)(\xi)$, we compute

$$\begin{aligned}
& T_m^{i+1}(V_0)(x) = \min\left\{x + \alpha T_m^i(V_0)(y_{m,1}),\ \max\{x, y_{m,1}\} + \alpha T_m^i(V_0)(0)\right\} F_m(y_{m,1}) \\
+ & \sum_{j=2}^{m} \min\left\{x + \alpha T_m^i(V_0)(y_{m,j}),\ \max\{x, y_{m,j}\} + \alpha T_m^i(V_0)(0)\right\} [F_m(y_{m,j}) - F_m(y_{m,j-1})]
\end{aligned}$$

for $x = 0$ and $x = \xi$. Then

$$\pi_m^{i+1}(x,\xi) \quad := \quad \begin{cases} 1 & \text{if} \quad x + \alpha T_m^{i+1}(V_0)(\xi) \geq \max\{x,\xi\} + \alpha T_m^{i+1}(V_0)(0) \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

Then it follows as in Proposition 9 that

$$\left\|V^* - T_m^{i+1}(V_0)\right\| \quad \leq \quad \frac{\varepsilon + \alpha^2\vartheta}{1-\alpha} \quad = \quad \frac{(1-\alpha)\epsilon/4 + \alpha^2(1-\alpha)\epsilon/(4\alpha^2)}{1-\alpha} \quad = \quad \frac{\epsilon}{2} \tag{9}$$

and from Proposition 10 that

$$\left\|V^* - V^{\pi_m^{i+1}}\right\| \quad \leq \quad 2\frac{\varepsilon + \alpha^2\vartheta}{1-\alpha} \quad = \quad 2\frac{(1-\alpha)\epsilon/4 + \alpha^2(1-\alpha)\epsilon/(4\alpha^2)}{1-\alpha} \quad = \quad \epsilon \tag{10}$$

with $\hat{T} = T_m$, $V = T_m^{i-1}(V_0)$, and $\hat{\pi} = \pi_m^{i+1}$.

Some features of these calculations are important. Note that for any x,

$$\begin{aligned} T_m(V)(x) \quad = \quad & \min\left\{x + \alpha V(y_{m,1}),\ \max\{x, y_{m,1}\} + \alpha V(0)\right\} F_m(y_{m,1}) \\ & + \sum_{j=2}^{m} \min\left\{x + \alpha V(y_{m,j}),\ \max\{x, y_{m,j}\} + \alpha V(0)\right\} \left[F_m(y_{m,j}) - F_m(y_{m,j-1})\right] \end{aligned}$$

Thus, to compute $T_m(V)(x)$ for any x, it is only necessary to evaluate the function V at the points $0, y_{m,1}, \ldots, y_{m,m}$. Thus one can calculate the sequence of functions $T_m^i(V) = T_m(T_m^{i-1}(V))$ by calculating, for each i, only the function values

$T_m^i(V)(0), T_m^i(V)(y_{m,1}), \ldots, T_m^i(V)(y_{m,m})$. Note that even though we calculate only these $m+1$ function values for each i, the functions $T_m^i(V)$ are well-defined at all $x \in [0,1]$, and they satisfy all the properties established above, and if desired $T_m^i(V)(x)$ can be computed for any $x \in [0,1]$ even if $T_m^{i-1}(V)$ was computed at only the points $0, y_{m,1}, \ldots, y_{m,m}$. One consequence of this observation is that the points $y_{m,1}, \ldots, y_{m,m}$ may be changed from iteration to iteration. For example, suppose that at each iteration i the points $y_{m_i,1}, \ldots, y_{m_i,m_i}$ are used. That is, the initial value function V_0 is calculated at points $0, y_{m_0,1}, \ldots, y_{m_0,m_0}$. Then

$$\begin{aligned} V_1(x) \quad := \quad & T_{m_0}(V_0)(x) \\ = \quad & \min\left\{x + \alpha V_0(y_{m_0,1}),\ \max\{x, y_{m_0,1}\} + \alpha V_0(0)\right\} F_{m_0}(y_{m_0,1}) \\ & + \sum_{j=2}^{m_0} \min\left\{x + \alpha V_0(y_{m_0,j}),\ \max\{x, y_{m_0,j}\} + \alpha V_0(0)\right\} \left[F_{m_0}(y_{m_0,j}) - F_{m_0}(y_{m_0,j-1})\right] \end{aligned}$$

is calculated at the points $x = 0, y_{m_1,1}, \ldots, y_{m_1,m_1}$. Proceeding by induction, suppose that $V_i(x) := T_{m_{i-1}}(V_{i-1})(x)$ has been calculated at the points $x = 0, y_{m_i,1}, \ldots, y_{m_i,m_i}$. Then

$$\begin{aligned}
V_{i+1}(x) &:= T_{m_i}(V_i)(x) \\
&= \min\{x + \alpha V_i(y_{m_i,1}),\ \max\{x, y_{m_i,1}\} + \alpha V_i(0)\} F_{m_i}(y_{m_i,1}) \\
&\quad + \sum_{j=2}^{m_i} \min\{x + \alpha V_i(y_{m_i,j}),\ \max\{x, y_{m_i,j}\} + \alpha V_i(0)\} [F_{m_i}(y_{m_i,j}) - F_{m_i}(y_{m_i,j-1})]
\end{aligned}$$

is calculated at the points $x = 0, y_{m_{i+1},1}, \ldots, y_{m_{i+1},m_{i+1}}$. It may speed up calculations to choose a different number m of points at different iterations; specifically, it may reduce computational effort to choose m_i to increase with i. One such method is given in the algorithm stated next, after we discuss some issues regarding the calculation of $\|V_i - V_{i-1}\|$.

Suppose that both $V_i(x) = T_m(V_{i-1})(x)$ and $V_{i-1}(x)$ have been calculated at the points $x = 0, y_{m,1}, \ldots, y_{m,m}$. Next, suppose that

$$\max_{x \in \{0, y_{m,1}, \ldots, y_{m,m}\}} |V_i(x) - V_{i-1}(x)| \quad \leq \quad \vartheta$$

Then it follows that

$$\|T_m(V_i) - V_i\| \quad := \quad \sup_{x \in [0,1]} |T_m(V_i)(x) - V_i(x)| \quad \leq \quad \alpha\vartheta \tag{11}$$

Let $\hat{V}_m$ denote the fixed point of T_m. Then, it follows as in Lemma 8 that

$$\left\|\hat{V}_m - T_m(V_i)\right\| \quad \leq \quad \frac{\alpha^2 \vartheta}{1 - \alpha}$$

The resulting policy π_m^{i+1} is given by (8), and the error is bounded by (9) and (10). Some additional properties of policy π follow from the earlier results. First, it follows from Lemma 18 in the appendix that for any $0 \leq x_1 \leq x_2 \leq 1$ it holds that

$$0 \quad \leq \quad T_m(V_i)(x_2) - T_m(V_i)(x_1) \quad \leq \quad x_2 - x_1$$

Let

$$\begin{aligned}
\zeta_m^{i+1}(\xi) &:= \xi + \alpha T_m(V_i)(0) - \alpha T_m(V_i)(\xi) \\
z_m^{i+1}(x) &:= \sup\{\xi \in [0,1] \ : \ x + \alpha T_m(V_i)(\xi) \geq \max\{x, \xi\} + \alpha T_m(V_i)(0)\}
\end{aligned}$$

We also have that for any $0 \leq \xi_1 \leq \xi_2 \leq 1$ it holds that

$$0 \quad \leq \quad \zeta_m^{i+1}(\xi_2) - \zeta_m^{i+1}(\xi_1) \quad \leq \quad \xi_2 - \xi_1$$

and for any $0 \leq x_1 \leq x_2 \leq 1$ it holds that

$$0 \quad \leq \quad z_m^{i+1}(x_2) - z_m^{i+1}(x_1) \quad \leq \quad \frac{x_2 - x_1}{1 - \alpha}$$

and, if $z_m^{i+1}(x_2) < 1$, then

$$x_2 - x_1 \quad \leq \quad z_m^{i+1}(x_2) - z_m^{i+1}(x_1) \quad \leq \quad \frac{x_2 - x_1}{1 - \alpha}$$

and that π_m^{i+1} satisfies, for any $x, \xi \in [0, 1]$,

$$\begin{aligned} \pi_m^{i+1}(x, \xi) \quad &= \quad \begin{cases} 1 & \text{if} \quad x \geq \zeta_m^{i+1}(\xi) \\ 0 & \text{otherwise} \end{cases} \\ &= \quad \begin{cases} 1 & \text{if} \quad \xi \leq z_m^{i+1}(x) \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

In words, whenever the algorithm stops, the resulting policy π_m^{i+1} is characterized by a threshold function $\hat{z}$ that is Lipschitz continuous with Lipschitz constant $1/(1-\alpha)$.

Algorithm for DSRP-I: Discounted Case.

Step 0 Choose $\epsilon > 0$ and $\epsilon_0 \geq \epsilon$. Set $\varepsilon_0 := (1-\alpha)\epsilon_0/4$ and $m_0 := \lceil 1/\varepsilon_0 \rceil$, and choose points $y_{m_0,1}, \ldots, y_{m_0,m_0}$, such that $y_{m_0,j} \in [(j-1)/m_0, j/m_0]$ for $j = 1, \ldots, m_0$. Choose an initial value function $V_0 : \{0, y_{m_0,1}, \ldots, y_{m_0,m_0}\} \mapsto \mathbb{R}$. Set $\epsilon_1 = \epsilon_0$, $m_1 = m_0$, $y_{m_1,j} = y_{m_0,j}$ for $j = 1, \ldots, m_1$. Set $i = 1$.

Step 1 Calculate $V_i(x) := T_{m_{i-1}}(V_{i-1})(x)$ at the points $x = 0, y_{m_i,1}, \ldots, y_{m_i,m_i}$.

Step 2 If $y_{m_i,j} = y_{m_{i-1},j}$ for $j = 1, \ldots, m_i$ and $\max_{x \in \{0, y_{m_i,1}, \ldots, y_{m_i,m_i}\}} |V_i(x) - V_{i-1}(x)| \leq \varepsilon_i/\alpha^2$ then go to Step 3. Otherwise, set $\epsilon_{i+1} = \epsilon_i$, $\varepsilon_{i+1} = \varepsilon_i$, $m_{i+1} = m_i$, $y_{m_{i+1},j} = y_{m_i,j}$ for $j = 1, \ldots, m_i$. Increment $i \leftarrow i + 1$ and go to Step 1.

Step 3 If $\epsilon_i \leq \epsilon$, then stop, with $T_{m_i}(V_i)$ approximating the optimal value function V^*, $\|V^* - T_{m_i}(V_i)\| \leq \epsilon/2$, and policy π_m^{i+1} given by (8) satisfying $\left\| V^* - V^{\pi_m^{i+1}} \right\| \leq \epsilon$.

Otherwise, set $\epsilon_{i+1} := \max\{\epsilon, 4\alpha^3 \max_{x\in\{0,y_{m_i,1},\ldots,y_{m_i,m_i}\}} |V_i(x) - V_{i-1}(x)| /(1-\alpha)\}$, $\varepsilon_{i+1} := (1-\alpha)\epsilon_{i+1}/4$, and $m_{i+1} := \lceil 1/\varepsilon_{i+1} \rceil$, and choose points $0, y_{m_{i+1},1}, \ldots, y_{m_{i+1},m_{i+1}}$, such that $y_{m_{i+1},j} \in [(j-1)/m_{i+1}, j/m_{i+1}]$ for $j = 1, \ldots, m_{i+1}$. Increment $i \leftarrow i+1$ and go to Step 1.

4.2.2 Finite Horizon Optimal Policy

In the previous section, we considered the DSRP-I with an infinite planning horizon. In Section 4.2.8, we compare the performance of various policies over a finite horizon, including the optimal value with perfect information, and in this section we briefly describe the DSRP-I with a finite horizon T.

The customer locations $\xi_t, t = 0, \ldots, T$ are independent, but not necessarily identically distributed. Let F_t denote the distribution function of ξ_t on $[0,1]$ with the discount factor $\alpha \in [0,1]$. We assume that the customer arriving in the final period T has to be served in that period. Thus the optimal value function V^* satisfies the following optimality equation:

$$
\begin{aligned}
V_T^*(x) &= \int_{[0,1]} \max\{x,\xi\}\, dF_T(\xi) \quad = \quad xF_T(x) + \int_{(x,1]} \xi \, dF_T(\xi) \\
V_t^*(x) &= \int_{[0,1]} \min\left\{x + \alpha V_{t+1}^*(\xi),\ \max\{x,\xi\} + \alpha V_{t+1}^*(0)\right\} dF_t(\xi), \\
&\quad t \in \{0,1,\ldots,T-1\} \qquad (12)
\end{aligned}
$$

It follows in the same way as for the infinite horizon case that for any $t \in \{0,1,\ldots,T\}$ and any $0 \le x_1 \le x_2 \le 1$ it holds that

$$0 \quad \le \quad V_t^*(x_2) - V_t^*(x_1) \quad \le \quad x_2 - x_1$$

Let

$$\zeta_t^*(\xi) \quad := \quad \xi + \alpha V_{t+1}^*(0) - \alpha V_{t+1}^*(\xi)$$

and

$$z_t^*(x) \quad := \quad \sup\left\{\xi \in [0,1] \ :\ x + \alpha V_{t+1}^*(\xi) \ge \max\{x,\xi\} + \alpha V_{t+1}^*(0)\right\}$$

Then an optimal policy is given by

$$\begin{aligned}\pi_t^*(x,\xi) &= \begin{cases} 1 & \text{if} \quad x \geq \zeta_t^*(\xi) \\ 0 & \text{otherwise} \end{cases} \\ &= \begin{cases} 1 & \text{if} \quad \xi \leq z_t^*(x) \\ 0 & \text{otherwise} \end{cases}\end{aligned}$$

The computational approach is similar to that discussed in Section 4.2.1. For example, for any points $y_1, \ldots, y_m$ such that $y_j \in [(j-1)/m, j/m]$ for $j = 1, \ldots, m$, and for each $t \in \{0, 1, \ldots, T\}$, let

$$\hat{F}_t(y) \;:=\; F_t(1/m)\mathbb{I}_{\{y \geq y_1\}} + \sum_{j=2}^{m} [F_t(j/m) - F_t((j-1)/m)]\mathbb{I}_{\{y \geq y_j\}}$$

and for any $V : [0,1] \mapsto \mathbb{R}$ and any $x \in [0,1]$, let

$$\begin{aligned}\hat{T}_t(V)(x) \;:=\;& \min\{x + \alpha V(y_1),\ \max\{x, y_1\} + \alpha V(0)\}\, \hat{F}_t(y_1) \\ &+ \sum_{j=2}^{m} \min\{x + \alpha V(y_j),\ \max\{x, y_j\} + \alpha V(0)\}\, [\hat{F}_t(y_j) - \hat{F}_t(y_{j-1})]\end{aligned}$$

For any $x \in [0,1]$, let k be such that $x \in [(k-1)/m, k/m]$. Then $\hat{V}_T(x) := xF_T(x) + y_k[\hat{F}_T(y_k) - F_T(x)] + \sum_{j=k+1}^{m} y_j[\hat{F}_T(y_j) - \hat{F}_T(y_{j-1})]$ and $\hat{V}_t(x) := \hat{T}_t(\hat{V}_{t+1})(x)$ for $t \in \{0, 1, \ldots, T-1\}$. Note that $\hat{V}_t(x)$ can be calculated for any given $x \in [0,1]$, even if $x \notin \{0, y_1, \ldots, y_m\}$, as long as $\hat{V}_{t+1}(\xi)$ has been calculated for $\xi \in \{0, y_1, \ldots, y_m\}$. Then the following algorithm can be used.

Algorithm for DSRP-I: Finite Horizon Case.

Step 0 Choose $\epsilon > 0$. Set $\varepsilon := \epsilon/(T+1)$ and $m := \lceil 1/\varepsilon \rceil$, and choose points $y_1, \ldots, y_m$, such that $y_j \in [(j-1)/m, j/m]$ for $j = 1, \ldots, m$. Calculate $\hat{V}_T(x)$ at the points $x = 0, y_1, \ldots, y_m$. Set $t = T - 1$.

Step 1 Calculate $\hat{V}_t(x)$ at the points $x = 0, y_1, \ldots, y_m$.

Step 2 The policy is given by

$$\hat{\pi}_t(x,\xi) \;:=\; \begin{cases} 1 & \text{if} \quad x + \alpha\hat{V}_{t+1}(\xi) \geq \max\{x, \xi\} + \alpha\hat{V}_{t+1}(0) \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

(Note that $\hat{\pi}_t(x, \xi)$ can be calculated for any given $x, \xi \in [0, 1]$.)

Step 3 If $t > 0$, then decrement $t \leftarrow t - 1$ and go to Step 1.

4.2.3 2-Stage Myopic Policy

In our 2-stage myopic policy, we consider only one future period when making a decision for the newly arriving customer. More specifically, if $x_t = 0$, then we delay serving the customer arriving at time t. That is, if there is no customer that has to be served at time t, then serving of the customer that arrives at time t is delayed. Otherwise, if $x_t > 0$, then we compare the cost of immediately serving (C_I) and the cost of delaying the service of (C_D) the newly arriving customer, where

$$C_I = max\{x_t, \xi_t\} + \xi_{t+1}$$

and

$$C_D = x_t + max\{\xi_t, \xi_{t+1}\}.$$

Because the location ξ_{t+1} is not yet known, we can only compare the expected values of these costs. If $E[C_I] \leq E[C_D]$, then the newly arriving customer is served immediately, otherwise it is delayed. Using the fact that ξ_{t+1} has distribution function F, we have $E[C_I] = max\{x_t, \xi_t\} + E[\xi_{t+1}]$ and $E[C_D] = x_t + E[max\{\xi_t, \xi_{t+1}\}]$, where $E[max\{\xi_t, \xi_{t+1}\}] = \int_0^{\xi_t} \xi_t dF(x) + \int_{\xi_t}^{\infty} x dF(x)$ and $E[\xi_{t+1}] = \int_0^{\infty} x dF(x)$. If F is $UNIF[0, 1]$, then $E[max\{\xi_t, \xi_{t+1}\}] = 0.5(1 + \xi_t^2)$ and $E[\xi_{t+1}] = 0.5$, which implies that we decide to serve immediately if $max\{x_t, \xi_t\} - x_t \leq 0.5\xi_t^2$ and to delay otherwise. (Note that we break the tie in favor of serving immediately.) This inequality certainly holds if $\xi_t \leq x_t$. If $\xi_t > x_t$, then we need to check if $\xi_t - x_t \leq 0.5\xi_t^2$ or equivalently if $(\xi_t - 1)^2 \geq 1 - 2x_t$. Note that for $x_t \geq 0.5$, the right hand side is non-positive, so that we choose to serve the customer immediately. If $x_t < 0.5$, then we choose to serve the customer immediately only if $\xi_t \leq 1 - \sqrt{1 - 2x_t}$. Combining these observations, the myopic policy for UNIF[0,1] reduces to the following simple rule: *If $x_t \geq 0.5$, then serve immediately; if $x_t < 0.5$, then serve immediately if $\xi_t \leq 1 - \sqrt{1 - 2x_t}$ and otherwise delay.*

Therefore, the myopic policy implies a threshold function $\zeta(x_t) = 1 - \sqrt{1 - 2x_t}$ for $x_t < 0.5$ and $\zeta(x_t) = 1$ for $x_t \geq 0.5$. If $\xi_t \leq \zeta(x_t)$, then we immediately serve the new customer, otherwise we delay its service to the next period.

4.2.4 Sampling-based Policy

The weakness of the 2-stage myopic policy is that it considers only one future period. In this section, we present a simple policy that does consider all future periods using Monte-Carlo sampling. The idea is to generate sample paths representing possible future arrivals and to use these sample paths when deciding whether to immediately serve or delay serving a newly arrived customer. The sampling-based policy is presented in Algorithm 3. Note that such sampling-based policies been been proposed by Bent and Van Hentenryck [66] under the umbrella of online stochastic optimization.

Algorithm 3 Sampling-based Policy

```
{Let N be the sample size.}
for t = 0 to T − 1 do
  if x_t = 0 then
    Delay serving ξ_t
  else
    for n = 1 to N do
      Create sample path ξ^n_{t+1}, ξ^n_{t+2}, ..., ξ^n_T
      Determine the optimal way to serve {x_t, ξ_t, ξ^n_{t+1}, ξ^n_{t+2}, ..., ξ^n_T}
      Record the decision for ξ_t, i.e., whether it is served immediately or whether its
        delayed
    end for
    if the number of times customer ξ_t is served immediately is greater than or equal to
      the number of times it is delayed then
      Serve ξ_t immediately
    else
      Delay serving ξ_t
    end if
  end if
end for
```

At each time t, the sampling-based policy does the following. If there is no customer that has to be served at time t, then the service of the customer arriving at time t is delayed. Otherwise, to decide whether to serve or to delay serving the customer arriving at time t, we create N random sample paths, where N is the sample size. Each sample path n consists of a set of locations $\xi^n_{t+1}, \xi^n_{t+2}, ..., \xi^n_T$ of future customer arrivals. Next, we determine the optimal way to serve the customers in the sample path. This is a deterministic problem that can be solved using a simple shortest path algorithm (see Section 4.2.1). We count the number of times the customer arriving at time t is served immediately, and the number of times its service is delayed. If the number of times the customer arriving at time t is served immediately is greater than or equal to the number of times its service is delayed, then we serve the customer immediately, otherwise we delay its service to time $t+1$.

4.2.5 Sample Average Approximation

Sample Average Approximation (SAA) is a method that can be used to solve stochastic programs. In this section, we present a multi-stage stochastic programming formulation of DSRP-I and show how SAA can be used to derive a policy.

The multi-stage stochastic programming formulation of DSRP-I is as follows:

$$\begin{aligned} \min_{u_0} \quad & \{c(x_0, \xi_0, u_0) \\ & + E_{\xi_1}\left[\min_{u_1} c(f(\xi_0, u_0), \xi_1, u_1) + E_{\xi_2}\left[\min_{u_2} c(f(\xi_1, u_1), \xi_2, u_2)\right.\right. \\ & \left.\left. + \cdots + E_{\xi_T}\left[\min_{u_T} c(f(\xi_{T-1}, u_{T-1}), \xi_T, u_T)\right] \cdots \right]\right]\Big\} \end{aligned}$$

where the location x_0 of the customer that has to be visited at time 0 and the location ξ_0 of the customer whose request arrives at time 0 are both given.

Before describing the k-stage SAA algorithm, we discuss the 2-stage SAA algorithm in more detail as the basic concepts can be presented more easily in this simpler case.

4.2.5.1 2-Stage SAA

In a 2-stage SAA approximation, we generate N independent and identically distributed sample paths of arrival locations $\xi^n_1, \xi^n_2, \ldots, \xi^n_T$ at times $1, 2, \ldots, T$ respectively from the distribution F. Then we solve the 2-stage problem

$$\begin{aligned} \min_{u_0} \quad & \{c(x_0, \xi_0, u_0) \\ & + \frac{1}{N}\sum_{n=1}^{N} \min_{\{u^n_1, u^n_2, \ldots, u^n_T\}} \left[c(f(\xi_0, u_0), \xi^n_1, u^n_1) + \sum_{t=2}^{T} c(f(\xi^n_{t-1}, u^n_{t-1}), \xi^n_t, u^n_t)\right]\Big\} \end{aligned}$$

Specifically, let

$$\bar{V}_0(x) \quad := \quad \frac{1}{N}\sum_{n=1}^{N} \min_{\{u_1^n, u_2^n, \ldots, u_T^n\}} \left[c(x, \xi_1^n, u_1^n) + \sum_{t=2}^{T} c(f(\xi_{t-1}^n, u_{t-1}^n), \xi_t^n, u_t^n)\right]$$

As mentioned before, in Section 4.2.7 we will show that for each x and each sample path $\xi_1^n, \xi_2^n, \ldots, \xi_T^n$,

$$\min_{\{u_1^n, u_2^n, \ldots, u_T^n\}} \left[c(x, \xi_1^n, u_1^n) + \sum_{t=2}^{T} c(f(\xi_{t-1}^n, u_{t-1}^n), \xi_t^n, u_t^n)\right]$$

can be computed by solving a shortest path problem on a simple acyclic network. If $c(x_0, \xi_0, 0) + \bar{V}_0(\xi_0) \geq c(x_0, \xi_0, 1) + \bar{V}_0(0)$, then we decide to visit the customer located at ξ_0 immediately, otherwise we decide to delay visiting that customer. Note that $\bar{V}_0(x)$ has to be computed for $x = 0$ and $x = \xi_0$ only.

4.2.5.2 k-Stage SAA

In the $k+1$-stage SAA approach a scenario tree is constructed inductively as follows. Let $\mathcal{N}^0 := \{0\}$ and $\xi_0^0 := \xi_0$. First we generate a set $\mathcal{N}_0^1$ of independent and identically distributed realizations of arrival locations $\xi_1^n, n \in \mathcal{N}_0^1$ at time 1 from the distribution F. Let $\mathcal{N}^1 := \left\{(0, n_1) : n_1 \in \mathcal{N}_0^1\right\}$. index the set of all histories ξ_0, ξ_1 up to time 1 in the sample. Given the set $\mathcal{N}^t$ of all histories $\xi_0, \ldots, \xi_t$ up to time t in the sample, we next generate the set $\mathcal{N}^{t+1}$ as follows. For each history $n^t \in \mathcal{N}^t$, we generate a set $\mathcal{N}_{n^t}^{t+1}$ of independent and identically distributed realizations of arrival locations $\xi_{t+1}^n, n \in \mathcal{N}_{n^t}^{t+1}$ at time $t+1$ from the distribution F. Since the sequence $\{\xi_t\}$ is independent, we can choose the same set $\mathcal{N}_{n^t}^{t+1}$ for each $n^t \in \mathcal{N}^t$, but we do not have to make such a choice. Then $\mathcal{N}^{t+1} := \left\{(n^t, n_{t+1}) : n^t \in \mathcal{N}^t, n_{t+1} \in \mathcal{N}_{n^t}^{t+1}\right\}$. Thus $\mathcal{N}^t$ is constructed for $t = 0, \ldots, k-1$. In addition, for each history $n^{k-1} \in \mathcal{N}^{k-1}$ we generate a set $\mathcal{N}_{n^{k-1}}$ of independent and identically distributed sample paths of arrival locations $\xi_k^n, \xi_{k+1}^n, \ldots, \xi_T^n$ at times $k, k+1, \ldots, T$ respectively from the distribution F.

Then the following $k+1$-stage problem is solved:

$$
\begin{aligned}
&\min_{u_0} \{c(x_0, \xi_0, u_0) \\
&+\frac{1}{|\mathcal{N}^1|} \sum_{n^1 \in \mathcal{N}^1} \left[\min_{u_1^{n^1}} c(f(\xi_0, u_0), \xi_1^{n^1}, u_1^{n^1}) \right. \\
&+\frac{1}{|\mathcal{N}^2_{n^1}|} \sum_{n^2 \in \mathcal{N}^2_{n^1}} \left[\min_{u_2^{n^2}} c(f(\xi_1^{n^1}, u_1^{n^1}), \xi_2^{n^2}, u_2^{n^2}) \right. \\
&+\cdots + \frac{1}{|\mathcal{N}^{k-1}_{n^{k-2}}|} \sum_{n^{k-1} \in \mathcal{N}^{k-1}_{n^{k-2}}} \left[\min_{u_{k-1}^{n^{k-1}}} c(f(\xi_{k-2}^{n^{k-2}}, u_{k-2}^{n^{k-2}}), \xi_{k-1}^{n^{k-1}}, u_{k-1}^{n^{k-1}}) \right. \\
&+\frac{1}{|\mathcal{N}_{n^{k-1}}|} \sum_{n \in \mathcal{N}_{n^{k-1}}} \min_{\{u_k^n, u_{k+1}^n, \ldots, u_T^n\}} \left[c(f(\xi_{k-1}^{n^{k-1}}, u_{k-1}^{n^{k-1}}), \xi_k^n, u_k^n) + \sum_{t=k+1}^{T} c(f(\xi_{t-1}^n, u_{t-1}^n), \xi_t^n, u_t^n) \right] \\
&\cdots]\}
\end{aligned}
$$

Specifically, for each $n^{k-1} \in \mathcal{N}^{k-1}$, let

$$
\bar{V}_{n^{k-1}}(x) \quad := \quad \frac{1}{|\mathcal{N}_{n^{k-1}}|} \sum_{n \in \mathcal{N}_{n^{k-1}}} \min_{\{u_k^n, u_{k+1}^n, \ldots, u_T^n\}} \left[c(x, \xi_k^n, u_k^n) + \sum_{t=k+1}^{T} c(f(\xi_{t-1}^n, u_{t-1}^n), \xi_t^n, u_t^n) \right]
$$

As before, $\bar{V}_{n^{k-1}}(x)$ can be computed by solving a number of shortest path problems on an acyclic network. Then, for each $t \in \{0, \ldots, k-2\}$ and each $n^t \in \mathcal{N}^t$, inductively define

$$
\bar{V}_{n^t}(x) \quad := \quad \frac{1}{|n \in \mathcal{N}^{t+1}_{n^t}|} \sum_{n \in \mathcal{N}^{t+1}_{n^t}} \min \left\{ c(x, \xi_{t+1}^n, 0) + \bar{V}_{(n^t,n)}(\xi_{t+1}^n),\ c(x, \xi_{t+1}^n, 1) + \bar{V}_{(n^t,n)}(0) \right\}
$$

Note that $\bar{V}_{n^t}(x)$ has to be computed for $x = 0$ and $x = \xi_t^{n^t}$ only. Also note that if for each $t \in \{0, \ldots, k-2\}$ the set $\mathcal{N}^{t+1}_{n^t}$ is the same for each $n^t \in \mathcal{N}^t$, and the set of sample paths $\mathcal{N}_{n^{k-1}}$ is the same for each $n^{k-1} \in \mathcal{N}^{k-1}$, then $\bar{V}_{(n^t,n)}$ does not depend on n^t, which reduces the computational effort significantly. In addition, note that

$$
\bar{V}_{n^t}(0) \quad = \quad \frac{1}{|n \in \mathcal{N}^{t+1}_{n^t}|} \sum_{n \in \mathcal{N}^{t+1}_{n^t}} \left[c(x, \xi_{t+1}^n, 0) + \bar{V}_{(n^t,n)}(\xi_{t+1}^n) \right]
$$

because it is always optimal to follow a decision to visit a customer immediately by a decision to delay the visit to the next customer. Finally, if $c(x_0, \xi_0, 0) + \bar{V}_0(\xi_0) \geq c(x_0, \xi_0, 1) + \bar{V}_0(0)$, then we decide to visit the customer located at ξ_0 immediately, otherwise we decide to delay visiting that customer.

4.2.6 Policies without Stochastic Future Information

As mentioned in the introduction, our research was motivated by the work of Angelelli et al. [2], in which the competitive ratio of a few online algorithms for DSRP-I was analyzed. To judge the value that knowledge about the future can bring, we include some of these online algorithms (policies) in our computational experiments. We introduce them briefly here.

The simplest dispatch policy is to alternate between serving the arriving customer immediately and delaying service to that customer to the next period. This policy, denoted by IDID, guarantees that every other period, two customers are served together. The IDID policy arises more naturally when the number of customers arriving in a period is greater than one. In fact, the policy is asymptotically optimal when the number of customers arriving goes to infinity.

Another natural dispatch policy, denoted by SMART(p), compares the cost of serving only the customer that must be served ($\xi_{t-1}u_{t-1}$) to the cost of serving the cost serving the customer that must be served together with the customer for which we have the option of delaying to the next period ($\max\{\xi_{t-1}u_{t-1}, \xi_t\}$). If the increase in cost is small, i.e., $\max\{\xi_{t-1}u_{t-1}, \xi_t\} \leq p \times \xi_{t-1}u_{t-1}$, then the customer for which we have an option is served immediately, otherwise it is delayed to the next period.

Note that SMART(p) is also a threshold policy, namely

$$s(x) = \begin{cases} px & \text{if} \quad x \leq \frac{1}{p} \\ 1 & \text{if} \quad \frac{1}{p} < x \leq 1. \end{cases}$$

4.2.7 Offline Optimal Solution

In the offline setting, we assume that customers are known beforehand so that there is perfect information. We use the value of the offline optimal solution in our computational experiments as a baseline for comparison.

An offline optimal solution can be computed by solving a shortest path problem on an acyclic graph. For each time $t = 1, 2, ..., T-1$, we create two nodes, representing the decision to serve the customer arriving in the period immediately and the decision to delay serving the customer arriving in that period. We also create a source node representing the decision to (immediately) serve the customer that has to be served in period 1 and a sink node representing the decision to (immediately) serve the customer that has to be served in period T. Arcs connecting nodes in subsequent periods represent the possible transitions, e.g., we can follow an immediate decision in period t with a delay decision in period $t+1$. The cost on the arcs represent the cost incurred in the period at the head of the arc. Note that an arc from a node representing a decision in period t to a node representing a decision in period $t+1$ captures two decisions, and that those two decisions completely define the cost in period $t+1$. For example, a decision to delay serving the customer arriving in period t and immediately serving the customer arriving in period $t+1$ completely determines the cost in period $t+1$ to be $max\{\xi_t, \xi_t + 1\}$. The minimum cost path defines an optimal solution.

4.2.8 Computational Study

Before analyzing and comparing the performance of the various dispatch policies on randomly generated finite horizon instances, we examine the optimal threshold policy for the infinite horizon case. In all our computational experiments we assume that the location of a newly arriving customer is uniformly distributed between 0 and 1.

The optimal threshold policy is computed using the following settings. First, we evaluate the recursive value function $T_m V(x)$ only at the mid points of $0, y_{m,1}, ..., y_{m,m}$, i.e., only for $x = 0.5 * y_{m,1}, 0.5 * (y_{m,1} + y_{m,2}), ..., 0.5 * (y_{m,m-1} + y_{m,m})$. Second, the number of discrete points at iteration i is set to $m_i = \lceil \frac{1}{\epsilon_i} \rceil$. Third, in Step 3 of the algorithm, if $\epsilon_i > \epsilon$, then ϵ_i is reduced and set to $0.5(\epsilon + \epsilon_i)$. As a consequence, the stopping criterion $\epsilon_i \leq \epsilon$ is modified to $\epsilon_i \leq \epsilon + \beta$, where β is a small constant.

Figure 28 shows the relevant portion of the optimal threshold functions for different discount factors. We only display the portion of the threshold function where the threshold value is less than 1. As soon as the threshold reaches 1, it will stay at 1. We see that the optimal threshold function is indeed nondecreasing and that the rate of increase is greater

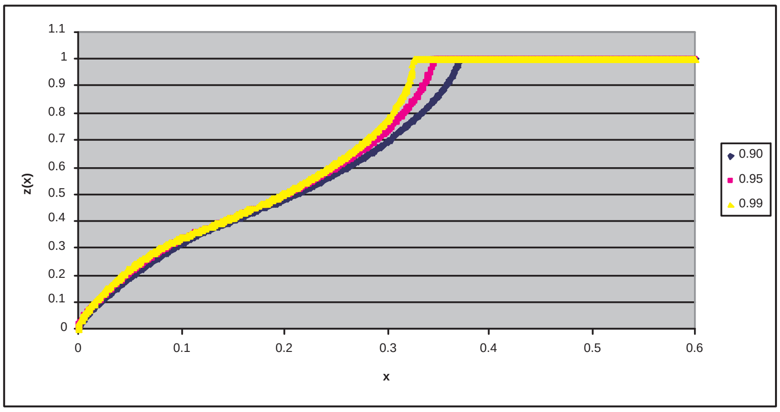

Figure 28: Threshold Functions for Optimal Policy on the Interval at Different Discount Factors

than 1 until the threshold value reaches its maximum value of 1. The graph also shows that when the location of the customer that has to be visited is farther than 0.33 from the depot (discount factor 0.99), the newly arriving customer should be served immediately. As the discount factor decreases, the threshold value $z^*(x)$ for x also decreases. This is to be expected, because as the discount factor decreases, the future becomes less important and we are more likely to delay.

Next, we compare the threshold functions of the optimal policy, the 2-stage myopic policy, and the SMART policy (for parameter values 1.5 and 2). These threshold functions are shown in Figure 29. We observe that the threshold function of the optimal policy is above

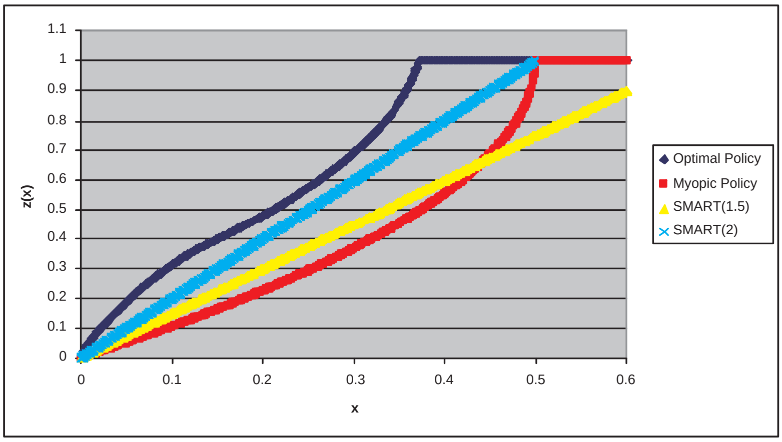

Figure 29: Threshold Functions for Optimal and Myopic Policies on the Interval

the threshold functions of the other policies. That is, when employing the optimal policy, we are more likely to serve a newly arriving customer immediately. The threshold function of SMART(2) is the next highest one. The detailed computational results to be presented next show that these two policies perform better than others. The threshold functions of the 2-stage myopic policy and SMART(1.5) are close to each for lower values of x, but for $0.45 \leq x \leq 0.67$ threshold function of myopic policy lies above that of SMART(1.5).

We compare the performance of the policies presented above on 30 randomly generated instances with a finite planning horizon of a 1000 periods. The cost incurred when applying each of the policies is shown in Table 16. The off-line optimal cost (Offline) provides a base line or lower bound. In addition, we present the cost for the finite horizon optimal policy (Optimal), the 2-stage myopic policy (Myopic), the k-stage SAA policy (SAA) with $k = 4$ and $\mathcal{N}_i^{i+1} = 10$, the sampling-based policy (Sampling) with $N = 2000$, two variants of the SMART online dispatch policy with parameters 1.5 and 2, and the IDID policy. As the k-stage SAA policy with $k = 4$ and $\mathcal{N}_i^{i+1} = 10$ for $i = 0, ...3$ requires the solution of 2000 shortest path problems in each period and the sampling-based policy also requires the solution of 2000 shortest path problems a comparison between them is fair. For convenience, we also provide the average cost over the 30 instances (AVE), and the percentage increase over the off-line optimal average cost (REL). Finally, we present the average run time for an instance in seconds.

We see that the finite horizon optimal policy has an average cost that is only 2.35% above the average off-line optimal cost. Furthermore, we see that both the sampling-based policy and the k-stage SAA policy do well, but that their computational requirements are substantially larger than those of the other policies due to the large number of shortest path computations. (Note that a linear-time shortest path algorithm was used since the underlying graph is acyclic.) As expected, SMART(2) performs reasonably well and the 2-stage myopic policy and SMART(1.5) perform relatively poorly.

Remark. Before moving on to variants of DSRP in which different assumptions are made regarding the location of the newly arriving customer, we observe that for DSRP-I, the

Table 16: Cost Comparison for Different Policies for DSRP-I

Instance	Offline	Optimal	Myopic	SAA	Sampling	SMART(1.5)	SMART(2)	IDID
1	623.05	637.81	650.08	640.53	637.42	653.30	639.98	655.05
2	630.43	648.81	663.94	653.94	649.98	669.20	652.11	669.85
3	647.93	665.46	680.20	673.66	665.12	683.07	670.27	683.95
4	631.22	647.39	657.27	651.96	648.80	657.81	648.53	672.82
5	629.22	642.42	657.17	647.73	644.64	660.82	646.65	662.85
6	623.97	638.48	654.83	641.90	640.91	654.78	641.06	665.85
7	639.33	654.61	667.26	656.59	656.67	671.13	657.84	672.29
8	644.69	660.57	670.18	659.68	660.58	674.68	660.37	681.76
9	630.75	642.15	657.07	646.41	647.35	662.95	647.37	675.04
10	640.83	656.43	672.33	659.46	655.36	671.28	663.07	668.56
11	632.55	646.05	660.55	650.87	646.40	659.23	649.63	671.70
12	649.00	660.58	675.13	664.97	661.64	678.75	664.48	693.49
13	614.94	633.74	646.83	630.56	633.39	645.67	638.53	658.28
14	624.74	638.98	652.28	640.26	643.93	654.66	639.95	663.34
15	629.70	644.30	661.25	649.14	646.03	660.55	647.74	672.43
16	641.22	654.83	667.11	659.15	657.67	674.81	659.08	668.71
17	637.04	650.45	667.50	656.46	651.84	669.60	651.53	678.21
18	624.06	634.83	652.38	638.49	635.69	654.01	638.96	670.48
19	625.26	638.77	650.63	641.41	641.10	657.47	639.96	661.53
20	617.91	632.05	643.19	634.02	635.66	643.63	634.47	659.44
21	639.11	659.87	666.44	659.33	662.96	669.18	660.06	681.04
22	634.28	649.90	662.37	656.12	653.75	665.47	652.83	679.95
23	626.58	642.03	656.84	646.19	639.44	657.77	644.58	666.54
24	642.65	653.42	676.40	658.99	657.40	679.24	659.72	675.02
25	630.14	646.28	661.07	650.80	646.71	665.95	646.45	666.60
26	633.26	651.82	657.03	653.43	650.14	663.91	653.86	672.71
27	627.89	639.51	658.55	644.10	639.36	659.42	645.21	665.75
28	607.72	620.06	641.70	626.41	621.64	648.67	629.40	638.41
29	627.55	643.73	659.31	648.06	644.07	656.19	643.31	662.52
30	630.90	647.15	659.66	649.90	647.72	667.42	651.81	664.41
AVE=	631.27	646.08	660.22	649.68	647.45	663.02	649.29	669.29
REL=	/	2.35	4.59	2.92	2.56	5.03	2.86	6.02
Run Time (sec)	1	1.4	1	114.4	219	1	1	1

assumption that only a single new customer arrives each period is irrelevant. When more than one new customer arrives in a period, it is sufficient to focus on the customer farthest away from the depot, say v. The reason is that if we decide to immediate serve v then we can serve the other newly arrived customers for free as we pass them on our way to v, and if we decide to delay serving v then we might as well delay serving the other newly arrived customers as we can serve them for free in the next period. This property does not hold for locations on the circle and locations on the disk.

4.3 DSRP on the Circle

In the Dynamic Stochastic Routing Problem on the Circle (DSRP-C) the setting is similar to the setting for DSRP-I except that each arrival is located on the unit circle, i.e., the circle with radius 1, instead of in the unit interval. Therefore, the location of the arrival at time t is denoted with the angle $\xi_t \in [0, 2\pi)$. The vehicle that serves customers starts and ends at the center of the circle. Travel occurs either along a ray from the center of the circle, or along a circular arc. Therefore, the cost of serving a single customer is 2 and the cost of serving two customers x and y is $2 + \min\{2, d(x, y)\}$, where $d(x, y) := \min\{(y - x) \bmod 2\pi,\ (x - y) \bmod 2\pi\} \in [0, \pi]$ denotes the minimum angle between x and y. The travel distance between x and y is equal to $\min\{2, d(x, y)\}$ because the vehicle can travel either through the center of the circle or along the circumference of the circle.

As before, $u_t \in \{0, 1\}$ denotes the decision at time t. Let

$$f(\xi, u) \quad := \quad \xi(1 - u) - u.$$

Then $x_t = f(\xi_{t-1}, u_{t-1}) \in \{-1\} \cup [0, 2\pi)$ denotes the location of the customer whose request was received at time $t-1$ and who has to be visited at time t; that is, if the customer whose request was received at time $t-1$ was visited at time $t-1$, then $x_t = -1$, and if the customer whose request was received at time $t - 1$ was not visited at time $t - 1$, then $x_t = \xi_{t-1}$. Let

$$c(x, \xi, u) \quad := \quad \begin{cases} 2u & \text{if} \quad x = -1 \\ 2 + \min\{2, d(x, \xi)\}u & \text{if} \quad x \in [0, 2\pi) \end{cases}$$

Then the cost incurred at time t is given by $c(x_t, \xi_t, u_t)$.

As before, we assume that x_0 is given, and that $\{\xi_t\}_{t=0}^{\infty}$ is an independent and identically distributed sequence with common distribution function F on $[0, 2\pi)$. Also, the objective is to minimize the expected total discounted cost over the planning horizon.

4.3.1 Infinite Horizon Optimal Policy

Let Π denote the set of all measurable functions $\pi : (\{-1\} \cup [0, 2\pi)) \times [0, 2\pi) \mapsto \{0, 1\}$ representing the stationary deterministic policies. Then the problem is given by (4), as for the DSRP-I. Also, the optimal value function $V^* : (\{-1\} \cup [0, 2\pi)) \mapsto \mathbb{R}$ is given by (5), as for the DSRP-I.

4.3.1.1 Properties of the Optimal Value Function and Optimal Policy

The optimal value function V^* satisfies the following optimality equation:

$$\begin{aligned} V^*(x) &= \mathbb{E}_F\left[\min_{u\in\{0,1\}} \{c(x,\xi,u) + \alpha V^*(f(\xi,u))\}\right] \\ &= \begin{cases} \int_{[0,2\pi)} \min\{\alpha V^*(\xi),\ 2+\alpha V^*(-1)\}\ dF(\xi) & \text{if} \quad x=-1 \\ \int_{[0,2\pi)} \min\{2+\alpha V^*(\xi),\ 2+\min\{2, d(x,\xi)\} + \alpha V^*(-1)\}\ dF(\xi) & \text{if} \quad x \in [0,2\pi) \end{cases} \end{aligned} \tag{14}$$

Since, for all x and ξ, $\alpha V^*(\xi) < 2+\alpha V^*(\xi)$ and $2+\alpha V^*(-1) \leq 2+\min\{2, d(x,\xi)\}+\alpha V^*(-1)$, it follows from (14) that $V^*(-1) \leq V^*(x)$ for all x.

Lemma 11 *$V^*(-1) = \int_{[0,2\pi)} \alpha V^*(\xi)\ dF(\xi)$, that is, if no customer remains from the previous time period, then it is optimal not to visit the newly arrived customer immediately, no matter where the new arrival is located.*

The proof of Lemma 11 establishes that for any $\xi \in [0, 2\pi)$, $\alpha V^*(\xi) < 2 + \alpha V^*(-1)$. Thus, if $d(x,\xi) \geq 2$, then $\min\{2, d(x,\xi)\} + \alpha V^*(-1) = 2 + \alpha V^*(-1) > \alpha V^*(\xi)$, and thus it is not optimal to visit the newly arrived customer immediately. Hence, travel between customers at x and ξ is optimal only if $d(x,\xi) < 2$, that is, travel between customers through the origin is never optimal.

Proposition 12 establishes that V^* is Lipschitz continuous with Lipschitz constant 1 (since the radius of the circle is 1). First we establish some properties of d. Recall that, for any $x, y, q \in \mathbb{R}$, if $x \bmod q + y \bmod q < q$, then $(x + y) \bmod q = x \bmod q + y \bmod q$. Note that if $x = y$, then $d(x, y) = (x - y) \bmod 2\pi = (y - x) \bmod 2\pi = 0$, and if $x \neq y$, then $(y - x) \bmod 2\pi + (x - y) \bmod 2\pi = 2\pi$, and thus $d(x, y) := \min\{(y - x) \bmod 2\pi, (x - y) \bmod 2\pi\} \leq \pi$ and $\max\{(y - x) \bmod 2\pi, (x - y) \bmod 2\pi\} \geq \pi$. Thus if $(x - y) \bmod 2\pi \leq \pi$, then $d(x, y) = (x - y) \bmod 2\pi$, and if $(y - x) \bmod 2\pi \leq \pi$, then $d(x, y) = (y - x) \bmod 2\pi$. Next we verify that d satisfies the triangle inequality. Consider any $x, y, z \in [0, 2\pi)$. We will show that $d(x, y) \leq d(x, z) + d(z, y)$. Without loss of generality, suppose that $d(x, y) = (x - y) \bmod 2\pi$. We have to consider the following 4 cases: (1) Suppose that

$d(x,z) = (x-z) \bmod 2\pi$ and $d(z,y) = (z-y) \bmod 2\pi$. Then $d(x,y) = (x-y) \bmod 2\pi = (x-z+z-y) \bmod 2\pi \leq (x-z) \bmod 2\pi + (z-y) \bmod 2\pi = d(x,z)+d(z,y)$. (2) Suppose that $d(x,z) = (z-x) \bmod 2\pi$ and $d(z,y) = (y-z) \bmod 2\pi$. Then $d(x,y) \leq (y-x) \bmod 2\pi = (y-z+z-x) \bmod 2\pi \leq (y-z) \bmod 2\pi + (z-x) \bmod 2\pi = d(x,z)+d(z,y)$. (3) Suppose that $d(x,z) = (x-z) \bmod 2\pi$ and $d(z,y) = (y-z) \bmod 2\pi$. Recall that $d(x,y) \leq \pi$ and $d(z,y) \leq \pi$. If $d(x,y) = \pi$ and $d(z,y) = \pi$, then $x = z$, and thus $d(x,y) = d(z,y) = d(x,z)+d(z,y)$. Otherwise, $d(x,y)+d(z,y) < 2\pi$, and thus $d(x,z) = (x-y+y-z) \bmod 2\pi = (x-y) \bmod 2\pi + (y-z) \bmod 2\pi = d(x,y)+d(z,y)$; hence, $d(x,y) = d(x,z)-d(z,y) \leq d(x,z)+d(z,y)$. (4) Suppose that $d(x,z) = (z-x) \bmod 2\pi$ and $d(z,y) = (z-y) \bmod 2\pi$. Recall that $d(x,z) \leq \pi$ and $d(x,y) \leq \pi$. If $d(x,z) = \pi$ and $d(x,y) = \pi$, then $y = z$, and thus $d(x,y) = d(x,z) = d(x,z)+d(z,y)$. Otherwise, $d(x,z)+d(x,y) < 2\pi$, and thus $d(z,y) = (z-x+x-y) \bmod 2\pi = (z-x) \bmod 2\pi + (x-y) \bmod 2\pi = d(x,z)+d(x,y)$; hence, $d(x,y) = d(z,y)-d(x,z) \leq d(x,z)+d(z,y)$.

Proposition 12 *For any $x_1, x_2 \in [0, 2\pi)$ it holds that*

$$|V^*(x_2) - V^*(x_1)| \quad \leq \quad d(x_1, x_2)$$

Next we establish additional properties of an optimal policy. Given the location $\xi \in [0, 2\pi)$ of the new arrival, let

$$\begin{aligned} \zeta^*(\xi) \quad &:= \quad \alpha\,[V^*(\xi) - V^*(-1)] \\ X^*(\xi) \quad &:= \quad \{x \in [0, 2\pi) \ : \ d(x,\xi) \leq \zeta^*(\xi)\} \end{aligned}$$

Proposition 13 *If the distance between the customer remaining from the previous time period located at $x \in [0, 2\pi)$ and the new arrival located at $\xi \in [0, 2\pi)$ is less than the threshold $\zeta^*(\xi)$, then it is optimal to visit the new arrival immediately, otherwise the service of the new arrival should be delayed.*

Note that $\alpha V^*(\xi) < 2 + \alpha V^*(-1)$ implies that $\zeta^*(\xi) < 2$. Also, consider any $\xi_1, \xi_2 \in [0, 2\pi)$. Then

$$|\zeta^*(\xi_2) - \zeta^*(\xi_1)| \quad = \quad \alpha\,|V^*(\xi_2) - V^*(\xi_1)| \quad \leq \quad \alpha d(\xi_1, \xi_2)$$

and thus ζ^* is Lipschitz continuous with Lipschitz constant α.

The optimality equation can be written as follows:

$$V^*(x) \quad = \quad \begin{cases} \int_{[0,2\pi)} \alpha V^*(\xi)\, dF(\xi) & \text{if} \quad x = -1 \\ \int_{\{\xi\in[0,2\pi)\,:\,d(x,\xi)\le\zeta^*(\xi)\}} [2 + d(x,\xi) + \alpha V^*(-1)] \;\, dF(\xi) & \\ + \int_{\{\xi\in[0,2\pi)\,:\,d(x,\xi)>\zeta^*(\xi)\}} [2 + \alpha V^*(\xi)] \;\, dF(\xi) & \text{if} \quad x \in [0, 2\pi) \end{cases}$$

As before, it may be more natural to characterize an optimal policy through the set of locations of the newly arriving customer such that the new customer should be visited immediately. Given $x \in \{-1\} \cup [0, 2\pi)$, let

$$\Xi^*(x) \quad := \quad \{\xi \in [0, 2\pi) \;:\; \alpha V^*(\xi) \ge \min\{2, d(x,\xi)\} + \alpha V^*(-1)\}$$

denote the set of locations of the newly arriving customer such that the new customer should be visited immediately. Lemma 11 established that $\Xi^*(-1) = \varnothing$. Let $z_+^*, z_-^* : [0, 2\pi) \mapsto [0, 2\pi)$ be given by

$$\begin{aligned} z_+^*(x) \quad &:= \quad \sup\{(\xi - x) \bmod 2\pi \;:\; \xi \in [0, 2\pi), d(x,\xi) = (\xi - x) \bmod 2\pi, \\ &\qquad\quad \alpha V^*(\xi) \ge \min\{2, d(x,\xi)\} + \alpha V^*(-1)\} \\ z_-^*(x) \quad &:= \quad \sup\{(x - \xi) \bmod 2\pi \;:\; \xi \in [0, 2\pi), d(x,\xi) = (x - \xi) \bmod 2\pi, \\ &\qquad\quad \alpha V^*(\xi) \ge \min\{2, d(x,\xi)\} + \alpha V^*(-1)\} \end{aligned}$$

Next we show that for all $x \in [0, 2\pi)$,

$$\Xi^*(x) \;\; = \;\; \left\{\xi \in [0, 2\pi) \;:\; (\xi - x) \mod 2\pi \le z_+^*(x)\right\} \bigcup \left\{\xi \in [0, 2\pi) \;:\; (x - \xi) \mod 2\pi \le z_-^*(x)\right\}$$

that is, if the previous customer is waiting to be served at $x \in [0, 2\pi)$, then it is optimal to serve the newly arrived customer located at $\xi \in [0, 2\pi)$ if the counterclockwise distance $(\xi - x) \bmod 2\pi \le z_+^*(x)$ or if the clockwise distance $(x - \xi) \bmod 2\pi \le z_-^*(x)$, and not to serve the newly arrived customer otherwise.

Proposition 14 *For all* $x \in [0, 2\pi)$*, it holds that* $\min\{2, d(x,\xi)\} + \alpha V^*(-1) \le \alpha V^*(\xi)$ *if* $(\xi - x) \bmod 2\pi \le z_+^*(x)$ *or* $(x - \xi) \bmod 2\pi \le z_-^*(x)$*, and* $\alpha V^*(\xi) < \min\{2, d(x,\xi)\} + \alpha V^*(-1)$ *otherwise.*

The proof of Proposition 14 shows that $z_+^*(x) < 2$ and $z_-^*(x) < 2$ for all x. It follows that it is never optimal to travel through the center of the circle between a previous arrival at x and a new arrival at ξ. Also, since $z_+^*(x) < 2$ and $z_-^*(x) < 2$ for all x, and for all $x \neq \xi$ it holds that $(\xi - x) \bmod 2\pi + (x - \xi) \bmod 2\pi = 2\pi$, it follows that if $x \neq \xi$ then it cannot hold that $(\xi - x) \bmod 2\pi \leq z_+^*(x)$ and $(x - \xi) \bmod 2\pi \leq z_-^*(x)$. It follows that the optimality equation (14) can be written as follows:

$$V^*(x) = \begin{cases} \int_{[0,2\pi)} \alpha V^*(\xi)\, dF(\xi); \text{ if; } x = -1 \\ \int_{\{\xi \in [0,2\pi) \,:\, (\xi - x) \bmod 2\pi \leq z_+^*(x)\}} [2 + (\xi - x) \bmod 2\pi + \alpha V^*(-1)]\ dF(\xi) \\ + \int_{\{\xi \in [0,2\pi) \,:\, \xi \neq x, (x-\xi) \bmod 2\pi \leq z_-^*(x)\}} [2 + (x - \xi) \bmod 2\pi + \alpha V^*(-1)]\ dF(\xi) \\ + \int_{\{\xi \in [0,2\pi) \,:\, (\xi - x) \bmod 2\pi > z_+^*(x),\ (x-\xi) \bmod 2\pi > z_-^*(x)\}} [2 + \alpha V^*(\xi)]\ dF(\xi); \text{ if; } x \in [0, 2\pi) \end{cases}$$

4.3.1.2 Example: Uniformly Distributed Points

For the special case of the DSRP-C in which ξ is uniformly distributed on the circle, it holds that $V^*(x) = V^*(0)$, for all $x \in [0, 2\pi)$ and $\zeta^*(\xi) = z_+^*(x) = z_-^*(x) = \zeta^*(0)$ for all $x, \xi \in [0, 2\pi)$.

It follows from the optimality equation (14) and Lemma 11 that

$$\begin{aligned} V^*(-1) &= \alpha V^*(0) \\ V^*(0) &= \frac{1}{2\pi} \int_0^{2\pi} \min\{2 + \alpha V^*(0),\ 2 + d(0,\xi) + \alpha V^*(-1)\}\ d\xi \\ &= \frac{1}{2\pi}\left[2\int_0^{\zeta^*(0)} \left\{2 + \xi + \alpha^2 V^*(0)\right\}\ d\xi + 2\int_{\zeta^*(0)}^{\pi} \{2 + \alpha V^*(0)\}\ d\xi\right] \\ &= \frac{1}{\pi}\left[2\zeta^*(0) + \frac{\zeta^*(0)^2}{2} + \alpha^2\zeta^*(0)V^*(0) + \{2 + \alpha V^*(0)\}\{\pi - \zeta^*(0)\}\right] \end{aligned}$$

It follows from the definition of ζ^* that $\zeta^*(0) = \alpha\left[V^*(0) - V^*(-1)\right] = \alpha(1-\alpha)V^*(0)$. Thus

$$\begin{aligned} V^*(0) &= \frac{1}{\pi}\Big[2\alpha(1-\alpha)V^*(0) + \frac{\alpha^2(1-\alpha)^2V^*(0)^2}{2} + \alpha^3(1-\alpha)V^*(0)^2 \\ &\qquad + 2\pi + \alpha\pi V^*(0) - 2\alpha(1-\alpha)V^*(0) - \alpha^2(1-\alpha)V^*(0)^2\Big] \\ \Rightarrow \quad V^*(0) &= \frac{-2\pi(1-\alpha) + \sqrt{4\pi^2(1-\alpha)^2 + 16\pi\alpha^2(1-\alpha)^2}}{2\alpha^2(1-\alpha)^2} \\ &= \frac{\sqrt{\pi^2 + 4\pi\alpha^2} - \pi}{\alpha^2(1-\alpha)} \\ \Rightarrow \quad \zeta^*(0) &= \frac{\sqrt{\pi^2 + 4\pi\alpha^2} - \pi}{\alpha} \end{aligned}$$

4.3.2 Finite Horizon Policies

The finite horizon policies can easily be modified to handle instances in which the location of the newly arriving customer is on the circle. We briefly comment on these modifications in this section.

The 2-stage myopic policy is a one-period look-ahead policy and thus involves $x_t \in \{-1\} \cup [0, 2\pi)$, the location of the customer that has to be served in period t, ξ_t the location of the customer that arrives in time period t, and ξ_{t+1} the location of customer that will arrive in time period $t+1$ with known distribution function F. If $x_t = -1$, serving of ξ_t is delayed. Otherwise, we compare the cost of immediately serving ξ_t and the cost of delaying serving ξ_t to the next period, where

$$C_I = [2 + min\{2, d(x_t, \xi_t)\}] + 2$$

and

$$C_D = 2 + [2 + min\{2, d(\xi_t, \xi_{t+1})\}].$$

If $E[C_I] \leq E[C_D]$, we choose to serve ξ_t immediately, else we choose to delay serving ξ_t to the next period. Since $d(x_t, \xi_t) \geq 2$ implies $E[C_I] \geq E[C_D]$, we choose to delay if $d(x_t, y_t) \geq 2$. Otherwise, we have to evaluate the inequality. Note that $E[min\{2, d(\xi_t, \xi_{t+1})\}] = \int_0^{2\pi} min\{2, d(\xi_t, \xi_{t+1})\} dF(x)$. This expression depends on ξ_t and F. If F is $UNIF[0, 2\pi)$, this expression is the same for any ξ_t. So, without loss of generality, we assume $\xi_t = 0$. We have

$$E[min\{2, d(0, \xi_{t+1})\}] = 2[\int_0^2 x \frac{1}{2\pi} dx + \int_2^\pi 2 \frac{1}{2\pi} dx] = \frac{2(\pi - 1)}{\pi}.$$

Substituting back into the inequality and using $d(x_t, \xi_t) \leq 2$, we choose to serve immediately if $d(x_t, \xi_t) \leq \frac{2(\pi-1)}{\pi}$. Therefore, the 2-stage myopic policy when the location of the newly arriving customer is uniformly distributed on the circle reduces to: *If $x_t = -1$, then choose to delay, else if $d(x_t, \xi_t) \leq \frac{2(\pi-1)}{\pi}$ then choose to serve immediately and otherwise choose to delay.*

The implementation of the finite horizon optimal policy is almost identical as we have used a state space discretization on the circle that is similar to the one we used for the interval.

The implementations of the k-stage SAA policy and the sampling-based policy as well as the policies that do not use any information about the future are adjusted by modifying the distance calculations. For example, for customers x_t and ξ_t on the interval, the cost of serving just x_t is x_t and serving them together is $max\{x_t, \xi_t\}$. For two customers x_t and ξ_t on the circle, the cost of serving just x_t is 2 and the cost of serving them together is $2 + \min(2, d(x_t, \xi_t))$.

4.3.3 Computational Study

Figure 30 shows the threshold angle for the infinite horizon optimal policy for various discount factors. We see that the threshold angle increases with the discount factor, i.e., we are more likely to immediately serve a newly arriving customer when the discount factor is higher. This is, again, expected since as the discount factor decreases, the future becomes less important and we are more likely to delay.

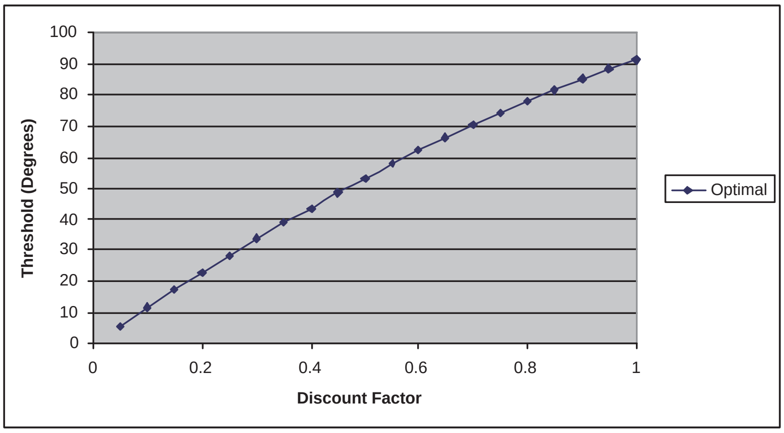

Figure 30: Optimal Threshold Angles on the Circle at Different Discount Factors

Next, we compare the performance of various dispatch policies. As before, we use a planning horizon of $T = 1000$ days. We create 30 problem instances, where a single customer arrives each period with a location uniformly distributed on the circle. The results can be found in Table 17.

Again, the finite horizon optimal policy performs well with an average cost that is only 2.46% above the average off-line cost. We see that in this setting the 2-stage myopic policy

Table 17: Cost Comparison for the Different Policies for DSRP-C

Instance	Offline	Optimal	Myopic	SAA	Sampling	SMART(1.5)	IDID
1	1569.66	1610.14	1620.15	1617.39	1666.71	1641.19	1678.28
2	1554.28	1587.17	1598.50	1590.59	1651.72	1632.34	1693.35
3	1537.74	1577.17	1586.82	1573.14	1641.83	1622.71	1655.23
4	1558.69	1599.49	1594.76	1604.81	1658.46	1640.12	1669.95
5	1551.35	1589.68	1599.62	1601.38	1651.34	1650.81	1675.69
6	1581.06	1617.21	1620.41	1620.59	1669.63	1663.57	1690.59
7	1580.42	1620.26	1625.29	1625.07	1684.71	1661.10	1711.17
8	1564.33	1606.64	1611.97	1608.26	1687.13	1652.99	1692.47
9	1544.84	1583.94	1591.72	1583.27	1662.32	1626.01	1679.17
10	1582.57	1624.64	1633.40	1628.93	1669.44	1657.24	1695.49
11	1555.91	1599.37	1611.96	1607.70	1673.73	1637.21	1682.62
12	1540.67	1574.13	1585.20	1579.20	1661.02	1621.30	1655.02
13	1562.57	1597.59	1603.67	1597.68	1669.48	1649.79	1703.17
14	1559.61	1594.91	1592.83	1593.54	1654.37	1628.01	1675.13
15	1564.15	1605.15	1601.72	1608.02	1670.43	1638.02	1680.39
16	1562.03	1597.83	1609.75	1598.87	1663.65	1639.93	1676.28
17	1552.52	1593.82	1605.04	1601.19	1657.42	1632.82	1685.05
18	1586.11	1619.36	1623.74	1621.54	1686.91	1668.91	1698.55
19	1550.68	1588.59	1598.08	1590.29	1647.56	1627.99	1670.51
20	1525.30	1563.08	1568.22	1566.21	1630.93	1602.08	1676.50
21	1545.39	1577.28	1586.31	1581.38	1639.13	1631.62	1668.18
22	1560.83	1601.31	1607.66	1610.43	1688.03	1645.88	1677.33
23	1565.19	1609.60	1613.98	1608.00	1655.78	1651.24	1684.35
24	1561.27	1596.83	1605.23	1600.49	1666.30	1633.03	1703.62
25	1548.71	1593.35	1592.95	1589.04	1641.05	1628.36	1677.41
26	1551.84	1590.55	1599.68	1595.99	1654.03	1631.88	1686.31
27	1532.23	1566.02	1571.20	1559.98	1632.13	1606.18	1679.96
28	1566.17	1607.66	1615.41	1608.37	1672.32	1657.61	1679.38
29	1563.07	1602.08	1610.22	1609.27	1665.99	1640.57	1684.88
30	1571.55	1604.87	1615.71	1610.30	1668.60	1644.44	1676.23
AVE=	1558.36	1596.66	1603.37	1599.70	1661.41	1638.83	1682.08
REL=	/	2.46	2.89	2.65	6.61	5.16	7.94
Run Time (sec)	1	1.6	1	243	297.8	1	1

performs well too with an average cost that is only 2.89% higher than average off-line cost. This is not that surprising when we compare the thresholds of the optimal policy and the 2-stage myopic policy, i.e., $\sqrt{\pi^2 + 4\pi\alpha^2} - \pi = 1.595$ versus $\frac{2(\pi-1)}{\pi} = 1.363$. Examining the thresholds also explains why the SMART policies do not perform well. For $p = 1.5$, the threshold is 1, and for $p = 2$ there is no threshold since SMART(2) is identical to the IDID policy. The k-stage SAA policy continues to perform well, but, surprisingly, the sampling-based policy performs poorly. The average run time for each of the policies increases slightly since we have to calculate the minimum angle between customers in order to decide whether to go through the center of the circle or along the circumference.

4.4 DSRP on the Disk

In the Dynamic Stochastic Routing Problem on the Disk (DSRP-D) the setting is similar to the setting for DSRP-I except that each arrival is located on the unit disk, i.e., the disk with radius 1, instead of in the unit interval. Therefore, the location of the arrival at time t is denoted with $\xi_t = (r_t, \theta_t) \in [0,1] \times [0, 2\pi)$, where r_t denotes the radius and θ_t denotes the angle of the location. Sometimes we will denote the radius and the angle of a location x with r_x and θ_x respectively, i.e., $x = (r_x, \theta_x)$. The vehicle that serves customers starts and ends at the center of the disk. Travel occurs either along a ray from the center of the disk, or along a circular arc. Consider any two locations x and y. Let $\theta_{xy} := \min\{\theta_x - \theta_y \mod 2\pi,\ \theta_y - \theta_x \mod 2\pi\} \in [0, \pi]$ denote the smallest angle between x and y. Consider the following four ways to travel between x and y (without loss of generality assume $r_x \geq r_y$):

1. Travel through the center of the disk, which gives travel distance between x and y of $r_x + r_y$.

2. Travel from x radially inwards towards the center of the disk to (r, θ_x) where $r \in (0, r_y)$. Then travel along a circular arc with radius r between (r, θ_x) and (r, θ_y). Finally, travel radially outwards from (r, θ_y) to y. This gives travel distance between x and y of $(r_x - r) + (r_y - r) + r\theta_{xy}$.

3. Travel from x radially inwards towards the center of the disk to (r_y, θ_x). Then travel along a circular arc with radius r_y between (r_y, θ_x) and $y = (r_y, \theta_y)$. This gives travel distance between x and y of $r_x - r_y + r_y\theta_{xy}$.

4. Travel from x radially inwards towards the center of the disk to (r, θ_x) where $r \in (r_y, r_x]$. Then travel along a circular arc with radius r between (r, θ_x) and (r, θ_y). Finally, travel radially inwards from (r, θ_y) to y. This gives travel distance between x and y of $(r_x - r) + (r - r_y) + r\theta_{xy}$.

Note that Option 3 is better than Option 4, because $r_x - r_y + r_y\theta_{xy} \leq (r_x - r) + (r - r_y) + r\theta_{xy}$ if $r > r_y$. Furthermore, either Option 1 or Option 3 is better than Option 2. If $\theta_{xy} > 2$, then Option 1 is better than Option 2, and if $\theta_{xy} \leq 2$, then Option 3 is better than Option 2. Therefore, the cost of visiting a single customer located at x is $2r_x$. The cost of visiting two customers located at x and y is $r_x + r_y + d(x,y)$, where $d(x,y) := \min\{r_x + r_y, \max\{r_x, r_y\} - \min\{r_x, r_y\}(1 - \theta_{xy})\}$ denotes the distance between x and y. Also, let $d'(x,y) := [r_x + r_y + d(x,y)] - 2r_x = r_y - r_x + d(x,y)$ denote the incremental distance for visiting both x and y over the distance for visiting only x. Note that, since $d(x,y) \leq r_x + r_y$, it follows that $d'(x,y) \leq 2r_y$.

As before, $u_t \in \{0, 1\}$ denotes the decision at time t. Let

$$f(\xi, u) \quad := \quad \xi(1-u) - u.$$

Then $x_t = f(\xi_{t-1}, u_{t-1}) \in \{-1\} \cup ([0,1] \times [0, 2\pi))$ denotes the location of the customer whose request was received at time $t-1$ and who has to be visited at time t; that is, if the customer whose request was received at time $t-1$ was visited at time $t-1$, then $x_t = -1$, and if the customer whose request was received at time $t-1$ was not visited at time $t-1$, then $x_t = \xi_{t-1}$. Let

$$c(x, \xi, u) \quad := \quad \begin{cases} 2r_\xi u & \text{if} \quad x = -1 \\ 2r_x + d'(x, \xi)u & \text{if} \quad x \in [0,1] \times [0, 2\pi) \end{cases}$$

Then the cost incurred at time t is given by $c(x_t, \xi_t, u_t)$.

We assume that x_0 is given, and that $\{\xi_t\}_{t=0}^{\infty}$ is an independent and identically distributed sequence with common distribution function F on $[0,1] \times [0, 2\pi)$.

4.4.1 Infinite Horizon Optimal Policy

Let Π denote the set of all measurable functions $\pi : (\{-1\} \cup ([0,1] \times [0, 2\pi))) \times ([0,1] \times [0, 2\pi)) \mapsto \{0, 1\}$ representing the stationary deterministic policies. Then the problem is given by (4), as for the DSRP-I. Also, the optimal value function $V^* : (\{-1\} \cup ([0,1] \times [0, 2\pi))) \mapsto \mathbb{R}$ is given by (5), as for the DSRP-I.

4.4.1.1 Properties of the Optimal Value Function and Optimal Policy

The optimal value function V^* satisfies the following optimality equation:

$$
\begin{aligned}
V^*(x) &= \mathbb{E}_F\left[\min_{u\in\{0,1\}} \{c(x,\xi,u)+\alpha V^*(f(\xi,u))\}\right] \\
&= \begin{cases} \int_{[0,1]\times[0,2\pi)} \min\{\alpha V^*(\xi),\ 2r_\xi+\alpha V^*(-1)\}\ dF(\xi) & \text{if} \quad x=-1 \\ \int_{[0,1]\times[0,2\pi)} \min\{2r_x+\alpha V^*(\xi),\ 2r_x+d'(x,\xi)+\alpha V^*(-1)\}\ dF(\xi) & \\ \text{if } x\in[0,1]\times[0,2\pi) & \end{cases}
\end{aligned} \tag{15}
$$

Note that $2r_\xi \leq r_x+r_\xi+d(x,\xi) = 2r_x+d'(x,\xi)$. Since, for all x and ξ, $\alpha V^*(\xi) \leq 2r_x+\alpha V^*(\xi)$ and $2r_\xi+\alpha V^*(-1) \leq 2r_x+d'(x,\xi)+\alpha V^*(-1)$, it follows from (19) that $V^*(-1) \leq V^*(x)$ for all x.

Lemma 15 $V^*(-1) = \int_{[0,1]\times[0,2\pi)} \alpha V^*(\xi)\ dF(\xi)$*, that is, if no customer remains from the previous time period, then it is optimal not to visit the newly arrived customer immediately, no matter where the new arrival is located.*

As was the case for DSRP-C, we expect it can be shown that there exists a threshold function $\eta(r,\theta)$ for all $r\in[0,1]$ and $\theta\in[0,2\pi)$ such that if the customer that has to be served is at (r_x,θ_x), then it is optimal to serve the newly arrived customer at (r_ξ,θ_ξ) if $D((r_x,\theta_x),(r_\xi,\theta_\xi)) \leq \eta(r_x,\theta_x)$, and not to serve the current customer otherwise, where $D(a,b)$ is the distance between points a and b on the disk. The derivation of the function $\eta(r_x,\theta_x)$ is left as future research.

4.4.2 Finite Horizon Policies

The finite horizon policies can easily be modified to handle instances in which the location of the newly arriving customer is on the disk. We briefly comment on these modifications in this section.

In the 2-stage myopic approach, if there is no customer that has to be served in period t, then we delay the service of the customer that arrives in time period t. Otherwise, we compare the expected cost of immediately serving and delaying the service of the newly arriving customer:

$$C_I = [r_{\xi_{t-1}}+r_{\xi_t}+\min\{r_{\xi_{t-1}}+r_{\xi_t},\max\{r_{\xi_{t-1}},r_{\xi_t}\}-(1-\theta_{\xi_{t-1}\xi_t})\min\{r_{\xi_{t-1}},r_{\xi_t}\}\}]+2r_{\xi_{t+1}}$$

and

$$C_D = 2r_{\xi_{t-1}} + [r_{\xi_t} + r_{\xi_{t+1}} + \min\{r_{\xi_t} + r_{\xi_{t+1}}, \max\{r_{\xi_t}, r_{\xi_{t+1}}\} - (1-\theta_{\xi_t\xi_{t+1}})\min\{r_{\xi_t}, r_{\xi_{t+1}}\}\}].$$

For $\theta_{\xi_{t-1},\xi_t} \geq 2$, we have $\min\{r_{\xi_{t-1}} + r_{\xi_t}, \max\{r_{\xi_{t-1}}, r_{\xi_t}\} - (1-\theta_{\xi_{t-1}\xi_t})\min\{r_{\xi_{t-1}}, r_{\xi_t}\}\} = r_{\xi_{t-1}} + r_{\xi_t}$, which implies that $C_I = 2(r_{\xi_{t-1}} + r_{\xi_t}) + 2r_{\xi_{t+1}}$ and $C_D = 2r_{\xi_{t-1}} + Cost(\xi_t, \xi_{t+1})$ where $Cost(\xi_t, \xi_{t+1})$ is the cost of serving customers ξ_t and ξ_{t+1} together. Since $Cost(\xi_t, \xi_{t+1}) \leq 2(r_{\xi_t} + r_{\xi_{t+1}})$, we choose to delay. If $\theta_{\xi_{t-1},\xi_t} < 2$ we have to evaluate the inequality $E[C_I] \leq E[C_D]$.

The joint probability density function for uniformly distributed customers on the unit disk is given by

$$f(x,y) = \begin{cases} \frac{1}{\pi}, & x^2 + y^2 \leq 1 \\ 0, & \text{otherwise} \end{cases}$$

where (x,y) denotes the cartesian coordinates of a location. Let R be the distance between a randomly selected point in the unit disk and the center of the disk. Since the point is selected randomly, equal area subsets of the disk should have equal probability, therefore for $0 < r' < 1$, the event $\{R \leq r'\}$ has probability $\frac{\pi r'^2}{\pi 1^2}$. So the cumulative distribution function of R becomes $F_R(r') = \begin{cases} 0, & r' < 0; \\ r'^2, & 0 \leq r' \leq 1; \\ 1, & 1 \leq r'. \end{cases}$ and the density function of R is

$$f_R(r') = \begin{cases} 2r', & 0 \leq r' \leq 1; \\ 0, & \text{else.} \end{cases} .$$

Using $f_R(r')$, $E[r_{\xi_{t+1}}] = \int_0^1 r' 2r' dr' = \frac{2}{3}$. Note that for a given $r_{\xi_{t+1}} = r'$, $\theta_{\xi_{t+1}}$ is distributed by $UNIF[0, 2\pi)$ and $E[Cost(\xi_t, \xi_{t+1})]$ depends on ξ_t only through r_{ξ_t}. So we may assume without loss of generality that $\theta_{\xi_t} = 0$. Thus,

$$\begin{aligned} E[Cost(\xi_t, \xi_{t+1})] &= 2\left[\int_0^{r_{\xi_t}} \int_0^2 (r_{\xi_t} + r + r_{\xi_t} - (1-\theta)r)\frac{1}{2\pi}2rdrd\theta \right. \\ &+ \left. \int_{r_{\xi_t}}^1 \int_0^2 (r_{\xi_t} + r + r - (1-\theta)r_{\xi_t})\frac{1}{2\pi}2rdrd\theta + \int_0^1 \int_2^{\pi} 2(r_{\xi_t} + r)\frac{1}{2\pi}2rdrd\theta\right] \\ &= \frac{2}{\pi}[\frac{8}{3}r_{\xi_t}^3 + \frac{4}{3}(1 - r_{\xi_t}^3) + r_{\xi_t}(1 - r_{\xi_t}^2) + (\pi - 2)(r_{\xi_t} + \frac{2}{3}r_{\xi_t}^3)] \end{aligned}$$

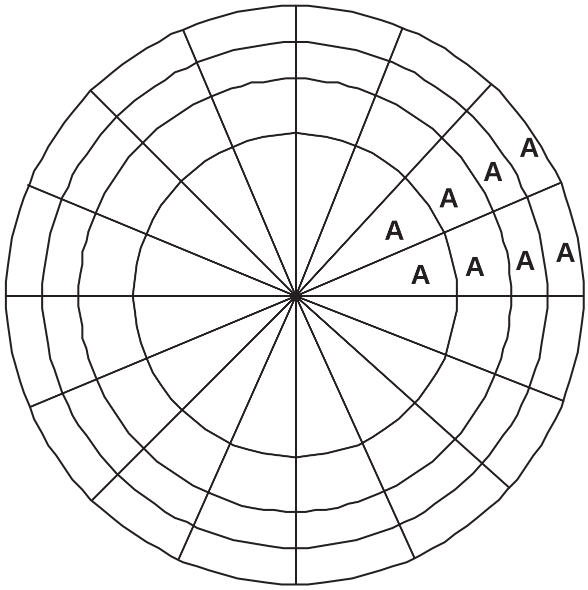

Figure 31: Uniform State Discretization on the Disk

Therefore, the 2-stage myopic policy reduces to: *If* $\theta_{\xi_{t-1}\xi_t} \geq 2$*, then choose to delay, else if* $r_{\xi_{t-1}} + r_{\xi_t} + \min\{r_{\xi_{t-1}} + r_{\xi_t}, \max\{r_{\xi_{t-1}}, r_{\xi_t}\} - (1 - \theta_{\xi_{t-1}\xi_t}) \min\{r_{\xi_{t-1}}, r_{\xi_t}\}\} + \frac{4}{3} \leq 2r_{\xi_{t-1}} + \frac{2}{\pi}[\frac{8}{3}r^3_{\xi_t} + \frac{4}{3}(1-r^3_{\xi_t}) + r_{\xi_t}(1-r^2_{\xi_t}) + (\pi-2)(r_{\xi_t} + \frac{2}{3}r^3_{\xi_t})]$*, then choose to serve immediately and otherwise choose to delay.*

The implementation of the finite horizon optimal policy requires modification as the discretization of the state space has to be done differently. In order to create a uniform discretization, we divide the disk into a number of regions with the same area as shown in Figure 31. Note that in this discretization scheme, we have two parameters: the number of regions k_1 covering the circle and the number of regions k_2 covering the radius. In Figure 31, we have shown a discretization with $k_1 = 16$ and $k_2 = 4$. The resulting state space has $k_1k_2 + 1$ states (it also includes state $\{-1\}$).

As before, the implementations of the k-stage SAA policy and the sampling-based policy as well as the policies that do not use any information about the future are adjusted by modifying the distance calculations.

4.4.3 Computational Study

We start by illustrating the 2-stage myopic policy. Recall that whenever there is no customer that has to be served, serving the newly arriving customer will be delayed. So we assume that there is a customer that has to be served. For three specific locations (r, θ), we show the region for which the 2-stage myopic policy decides to serve the newly arriving customer immediately (Figures 32, 33, and 34). Two observations can be made. First, the region is symmetric with respect to the radial line that emanates from the depot and passes through the customer that has to be served (which is to be expected due to the fact that we assume that θ is uniformly distributed in $[0, 2\pi)$). Second, as the radial distance r increases, the size of the region increases as well.

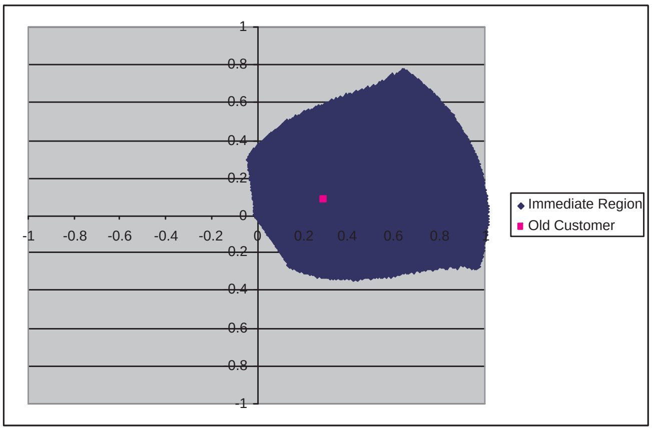

Figure 32: Customer Region to be Served Immediately: Example 1

Next, we compare the performance of the various dispatch policies. As before, we use a planning horizon of $T = 1000$ days. We create 30 problem instances, where a single customer arrives each period with a location uniformly distributed on the disk. We use rejection sampling to generate uniformly distributed locations on the disk. We generate x and y independently from UNIF[-1,1]. If (x, y) falls within the disk, i.e., if $x^2 + y^2 \leq 1$, then we accept, otherwise we reject (see Figure 35). The results of the experiments can be found in Table 18.

We see that the finite horizon optimal policy and the 2-stage myopic policy perform well with averages of 2.16% and 2.14% above the average cost of off-line optimal solutions. The

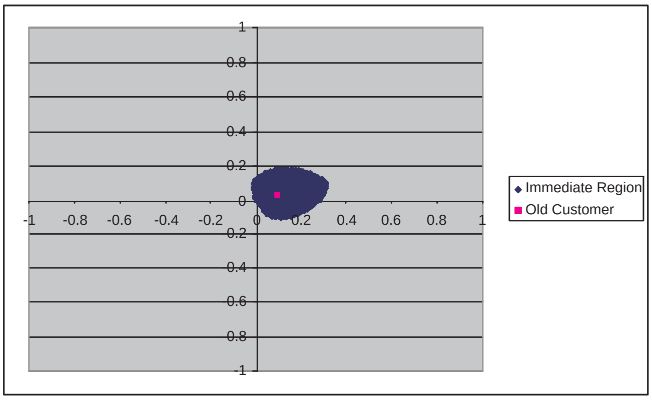

Figure 33: Customer Region to be Served Immediately: Example 2

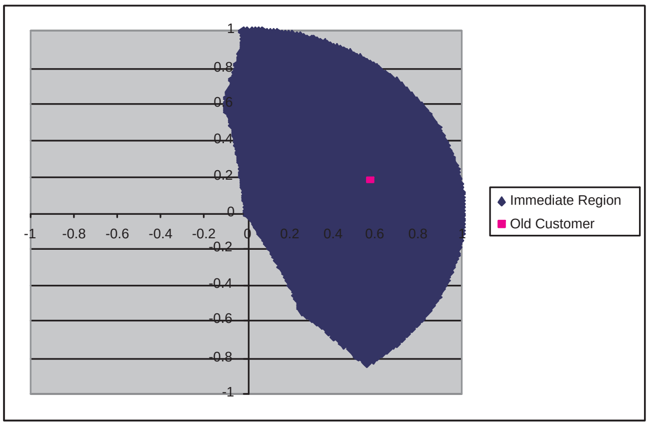

Figure 34: Customer Region to be Served Immediately: Example 3

fact that the 2-stage myopic policy sometimes outperforms the finite horizon optimal policy is a result of the discretization used. The k-stage SAA policy outperforms the sampling-based policy and provides solutions of reasonable quality. Among the policies that do not use any information about the future, SMART(2) does best.

4.5 Multiple Customers

So far, we have assumed that a single customer arrives in each period. In this section, we will consider the variant in which multiple customers arrive in each period and their locations are in the Euclidean plane. This variant much more closely resembles situations that arise in practice. Of course, determining the cost incurred on a particular day becomes

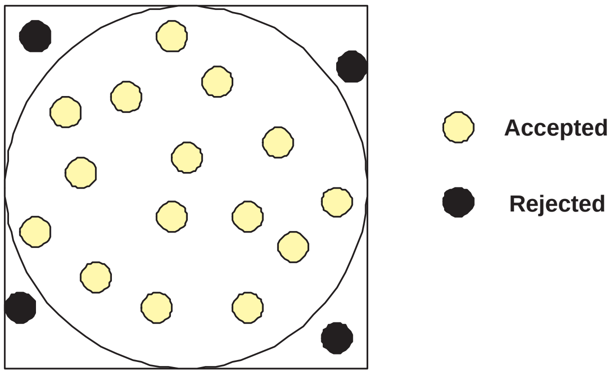

Figure 35: Rejection Sampling to Create Uniformly Distributed Customers on the Disk

much more difficult as it involves solving a TSP.

When multiple customers arrive in a period, the problem not only has a routing component, but also has a partition component. The set of customers arriving in a period has to be partitioned into a subset of customers that will be served immediately and a subset of customer for which serving is delayed until the next period.

Of course the complexity of computing the cost incurred in a period given the set of customers to be served makes solving the off-line optimization problem much harder, but that is not all. The network representing the off-line optimization problem no longer has polynomial size. The off-line optimization problem can be solved in the same way as before, but the number of nodes required to represent each period is exponential in the number of customers that arrive in that period.

In the remainder, we will discuss how we have adapted some of the dispatching policies discussed earlier to this new situation. In this discussion, we assume the existence of a "black box" or "oracle" for the solution of a TSP.

4.5.1 Heuristic Implementations of SMART(p)

Let C_t be the set of customers that has to be served at time period t. Let $C_{t|t+1}$ be the set of customers that may be served either at time period t or $t+1$, for $t = 1, ..., T-1$. We provide two different implementations of SMART(p):

Implementation 1: At time period t, consider the TSP tour that serves the customers in C_t, say T_1 with cost $c(T_1)$. For each customer i in $C_{t|t+1}$, we compute the increase in the

Table 18: Total Cost Comparison between Different Algorithms on the Disk

Instance	Offline	Online	Myopic	SAA	Sampling	SMART(1.5)	SMART(2)	IDID
1	1095.45	1124.73	1120.36	1126.73	1137.45	1165.35	1152.77	1181.54
2	1088.88	1111.37	1107.98	1111.94	1126.17	1161.31	1140.56	1162.78
3	1079.61	1106.67	1108.52	1107.87	1117.05	1148.15	1144.05	1157.81
4	1109.96	1135.39	1134.10	1136.00	1154.17	1174.72	1165.19	1193.04
5	1090.60	1112.38	1111.33	1114.67	1125.69	1154.07	1145.89	1148.38
6	1107.86	1137.31	1137.04	1133.08	1146.82	1181.40	1152.17	1179.01
7	1102.71	1125.53	1127.36	1126.11	1140.16	1163.90	1153.16	1174.73
8	1106.98	1127.14	1127.46	1130.44	1144.07	1172.25	1163.19	1185.67
9	1078.64	1100.52	1104.58	1100.80	1116.86	1146.27	1130.33	1156.30
10	1106.40	1131.51	1129.53	1132.12	1143.44	1180.49	1163.14	1177.12
11	1077.97	1104.09	1103.01	1105.77	1117.88	1153.54	1124.49	1158.44
12	1082.89	1105.35	1105.62	1107.82	1119.85	1149.92	1130.83	1158.53
13	1078.68	1102.68	1104.03	1106.44	1118.89	1147.13	1133.40	1160.28
14	1091.89	1111.20	1113.30	1115.53	1130.98	1159.07	1145.78	1161.25
15	1091.51	1115.86	1117.12	1117.07	1124.82	1159.36	1150.22	1155.93
16	1101.66	1126.20	1124.02	1123.46	1138.31	1169.90	1158.04	1169.83
17	1075.02	1094.36	1094.66	1095.92	1119.64	1145.69	1122.34	1147.55
18	1134.29	1155.84	1158.06	1159.03	1175.11	1196.49	1180.97	1202.00
19	1074.98	1098.58	1093.38	1098.93	1113.69	1142.50	1125.93	1144.43
20	1102.15	1126.95	1130.64	1131.42	1142.58	1162.66	1150.80	1172.60
21	1092.27	1121.16	1121.12	1119.45	1131.31	1155.49	1137.27	1151.63
22	1091.01	1116.74	1112.15	1118.06	1127.73	1161.30	1155.36	1158.82
23	1104.02	1124.02	1123.19	1127.55	1140.99	1171.88	1159.67	1178.64
24	1121.15	1141.10	1142.70	1141.67	1151.70	1185.68	1172.15	1195.22
25	1087.07	1107.36	1106.86	1112.84	1120.34	1157.00	1144.19	1154.66
26	1080.93	1104.95	1102.52	1107.54	1113.84	1149.31	1136.09	1137.58
27	1091.97	1114.51	1112.14	1113.62	1129.25	1165.63	1142.47	1159.04
28	1085.56	1109.97	1112.18	1111.46	1118.47	1153.12	1140.46	1157.09
29	1099.35	1125.25	1125.83	1124.60	1130.89	1173.00	1146.50	1162.25
30	1102.71	1125.00	1126.05	1130.23	1142.39	1166.21	1163.71	1178.87
AVE=	1094.47	1118.12	1117.89	1119.61	1132.02	1162.43	1147.70	1166.03
REL=	/	2.16	2.14	2.30	3.43	6.21	4.86	6.54
Run Time (sec)	1	102.1	1	333.5	812.7	1	1	1

cost of T_1 if customer i is added to T_1 using, for example, cheapest insertion, say I_i. Next, we identify the customer i with the smallest I_i and, if $c(T_1) + I_i \leq p \times c(T_1)$, we insert it into T_1. The process repeats as long as customers can be added to T_1 with the following set up, we compare the *current cost of* T_1 plus the *current insertion cost of* i with p times the *initial cost of* T_1.

Implementation 2: At time period t, consider the two TSP tours that serve the customers in C_t and in $C_{t|t+1}$, say T_1 with cost $c(T_1)$ and T_2 with cost $c(T_2)$. For each customer i in T_2, we compute the decrease in the cost of T_2 if customer i is removed, say

D_i, and we compute the increase in the cost of T_1 if customer i is added to T_1 using, for example, cheapest insertion, say I_i. Next, we identify the customer i with the largest value of $D_i - I_i$, remove it from T_2 and insert it into T_1 if $c(T_1) + I_i \leq p \times c(T_1)$. Again, the process repeats as long as customers can be added to T_1 with the following set up, we compare the *current cost of* T_1 plus the *current insertion cost of* i with p times the *initial cost of* T_1.

In our actual implementations we use Algorithm 4 to solve TSPs.

Algorithm 4 TSP Heuristic

1. Find the furthest customer from depot. Let's call it customer x. Create the route 0-x-0, with 0 representing the depot
2. Find the cheapest insertion cost for each un-inserted customer into the current route. Choose the one with the smallest insertion cost and insert it into current route. Repeat **2** until all customers are inserted. (Note that the insertion costs are updated after each insertion.)
3. Using the order of insertion established in **2**, remove and re-insert each customer using cheapest insertion
4. Repeat **3** as long as there is an improvement in the cost

A complete description of SMART(p), based on the second implementation, is given in Algorithm 5.

Algorithm 5 Smart(p)

$t \leftarrow 1$
while $t \leq T - 1$ **do**
  Find tours T_t and $T_{t|t+1}$ for customers in C_t and $C_{t|t+1}$.
  $L \leftarrow c(T_t)$
  repeat
    For i in $T_{t|t+1}$ compute D_i, the cost of deleting i from $T_{t|t+1}$.
    For i in $T_{t|t+1}$ compute I_i, the cheapest insertion cost of i into T_t.
    Find i in $T_{t|t+1}$ with maximum value of $D_i - I_i$.
    if $c(T_t) + I_i \leq p * L$ **then**
      Delete i from $T_{t|t+1}$.
      Insert i into T_t
    end if
  until $T_{t|t+1} = \emptyset$ or $c(T_t) + I_i > p * L$
  if $T_{t|t+1} \neq \emptyset$ **then**
    $C_{t+1} = T_{t|t+1}$
    $t \leftarrow t + 1$
  else
    $C_{t+2} \leftarrow C_{t+1|t+2}$
    $t \leftarrow t + 2$.
  end if
end while

Table 19 shows the performance of SMART(p) for two different values of parameter p on a set of 36 instances. Each instance has 40 periods. The number of customers arriving in each period is drawn from a discrete uniform distribution over the interval [*lower*, *upper*]. The number of customers arriving in each period is referred to as the density of an instance. The density of an instance is high if $\frac{lower+upper}{2}$ is large and the variability of the density is high if $upper - lower$ is large. Each arriving customer has a location with an x-coordinate and y-coordinate drawn from a continuous uniform distribution over the interval [-1,1]. Thus the instances differ both in terms of density and in terms of the locations of the customers.

We see that the performance of SMART(p) seems to improve as p increases. We also see that the average cost for policy IDID is better than the one for SMART(2). In fact, the average cost of policy IDID is better than the one for SMART(3) as well; the average cost of SMART(3) is 186.095. This suggests that the policy IDID may be close to optimal in this setting, especially for high-density instances with uniformly distributed locations.

Table 20 shows the average cost of both SMART(p) implementations over the 36 instances. We see that the average cost for SMART(p) with the second implementation is only slightly better than with the first implementation.

4.5.2 A Sampling-based Policy

If we know the distribution of the location of future customers, we can use sampling to improve our decision-making. At the beginning of time period t, we know the set of customers that have to be served, i.e. C_t, and the set of newly arriving customers that may be served either in time period t or in time period $t+1$, i.e. $C_{t|t+1}$. Now suppose that the number of customers arriving in a period is given by a discrete uniform distribution on the interval $[lower, upper]$ and that the location of these customers is uniformly distributed on the unit square. We can now sample $C_{t+1|t+2}$. For each sample, we now decide whether to shift customers from $T_{t|t+1}$ into T_t *or* into $T_{t+1|t+2}$, using cheapest insertion. We can repeat this process for N samples and then use some rule to decide what to do with each of the customers in the set $C_{t|t+1}$. Note that the suggested sampling-based policy looks one period ahead. A detailed description of the procedure can be found in in Algorithm 6. Note that the selection criterion $\frac{CI_j}{I_j} \leq \frac{CD_j}{D_j}$ can be changed to $I_j \geq D_j$ if we want to use simple counting instead of an average cost comparison to decide whether to serve customer $j \in C_{t|t+1}$ immediately or to delay serving until the next period. Note also that we create UNIF$(lower, upper)$ customers for each period.

Table 21 shows the results we have obtained using the sampling-based policy using both a simple counting and an average cost criterion. Note that the sampling-based policy is outperformed by the simple IDID solution for multi-customer per period case when customers are uniformly distributed. This is somewhat counter-intuitive as we use information about future orders in the sampling-based policy. However, the results may be partially explained by the density. The higher the density, the more likely it is that "batching" customers, i.e., serving customers every other day, is an effective policy and this is exactly what the IDID policy does. Therefore, we next examine the performance of the sampling-based policy on very low-density instances. The results can be found in Table 22. (Note that because the density is so small, we can also compute an off-line solution.) We see that the sampling-based policy still does not perform great, but it does perform better than the IDID policy.

4.5.3 Non-uniformly Distributed Customers.

Up to now, we have not considered non-homogenous distributions of locations of arriving customers. However, as can be expected, when the locations of arriving customers are no longer uniformly distributed, the value of future information becomes more important and algorithms that use future customer distribution information explicitly start to perform better than those that do not take this information into account.

Table 23 shows 16 different probability distributions used to create customer locations that are not uniform. Customers are uniformly distributed over k small squares each with a given width (represented in the table in the form of half this width). We create UNIF$[lower, upper]$ customers each period and the percentage of these customers appearing in each of the small squares is also given. For example, under Setting 5, we create two

Algorithm 6 A Sampling-based Policy

$t \leftarrow 1$,
while $t \leq T - 1$ **do**
  $T_t \leftarrow$ TSP tour through customers in C_t
  $T_{t+1} \leftarrow$ TSP tour through customers in $C_{t|t+1}$
  $I_j \leftarrow 0\ \forall j \in C_{t|t+1}$
  $D_j \leftarrow 0\ \forall j \in C_{t|t+1}$
  $CI_j \leftarrow 0\ \forall j \in C_{t|t+1}$
  $CD_j \leftarrow 0\ \forall j \in C_{t|t+1}$
  for $s = 1$ to N **do**
    Generate sample $C_{t+1|t+2}$
    $T^s_{t+1|t+2} \leftarrow$ TSP tour through customers in sample
  end for
  for $s = 1$ to N **do**
    Restore tours T_t and $T_{t|t+1}$.
    For each customer in $T_{t|t+1}$ find cheapest insertion into T_t and into $T^s_{t+1|t+2}$.
    Find the best customer to insert into T_t, say a. Find the best customer to insert into $T^s_{t+1|t+2}$, say b. If the insertion cost for a into T_t is less than or equal to the insertion cost of customer b into $T^s_{t+1|t+2}$, then insert customer a into T_t; otherwise insert customer b into $T^s_{t+1|t+2}$.
    Update insertion costs and continue until tour $T_{t|t+1}$ has no more customers. Record for each customer $j \in C_{t|t+1}$ whether it was inserted into T_t or into $T^s_{t+1|t+2}$. If j was inserted into T_t, increment I_j by one and increment CI_j by the cost of modified tours T_t and $T^s_{t+1|t+2}$. Otherwise, increment D_j by one and increment CD_j by the cost of modified tours T_t and $T^s_{t+1|t+2}$.
  end for
  for $j \in C_{t|t+1}$ **do**
    if $\frac{CI_j}{I_j} \leq \frac{CD_j}{D_j}$ **then**
      Serve j immediately
    else
      Delay serving j
    end if
  end for
  Find TSP tour through customers in C_t and those customers in $C_{t|t+1}$ that are to be served immediately.
  if there are customers in $C_{t|t+1}$ that are to be delayed **then**
    $C_{tmp} \leftarrow$ customers $C_{t|t+1}$ that are to be delayed
    $t \leftarrow t + 1$
    $C_t \leftarrow C_{tmp}$
  else
    $C_{t+2} \leftarrow C_{t+1|t+2}$
    $t \leftarrow t + 2$
  end if
end while

small squares with their centers randomly chosen, one with half-width 0.1 and one with half-width 0.2. The first of these squares contains 66% of total number of customers and the second contains 34%.

Table 24 summarizes the average costs under the IDID policy and the sampling-based policy for the different settings. Note that for each of these settings, we have again generated 36 instances using the same density characteristics as before, i.e., lower and upper bounds on the number of customers arriving in each period. The cost values shown in Table 24 for each setting are the averages over these 36 instances.

We see that the sampling-based policy now does on average 1.59% better. On the other hand, it does not always provide the best solution. For example, the average costs are worse for Setting 15 and Setting 16. However, the characteristics of these settings are such that the probability distributions may not be too far from uniform.

4.6 Concluding Remarks

We have considered several variants of a dynamic stochastic routing problem, in which the location of future customers is only know in distribution. For each of these variants, we have derived optimal policies and we have conducted computational studies in which various dispatch policies were compared, some of which do not use the information regarding the location of future customers, to analyze their quality as well as their efficiency. The optimal policies all turn out to be threshold polices, and, not surprisingly, our computational experiments show that using information about the future is beneficial. We have also made initial steps towards extending the problem setting to one that more closely resembles real-life situations, i.e., in which multiple customers arrive per period, but much work remains to be done for that setting.

A couple of final interesting observations can be made regarding the results of computational experiments. First, myopic policy seems to perform reasonably well. This is due to the problem setting considered: we have a delivery guarantee of two days, and myopic policy is considering only one future period. However, if delivery agreement would be more than two days, it is highly likely that a policy which considers the cost of only two time periods would not perform as well. To see this, suppose delivery guarantee is three days. So, at the beginning of a time period t, we have three (possible) set of customers:

A: Customers that have to be served at time t (two days old customers)

B: Customers that must be served either at time t or $t+1$. (one day old customers)

C: Customers that must be served either at time t, $t+1$ or $t+2$ (new customers)

Now, while deciding when to serve customers in sets B and C, using a policy that considers the costs of only two time periods would certainly not capture the problem well. In fact, for period $t+1$, it is not clear how to include costs associated with potential postponed customers in set C. Second, sampling policy and SMART(2) perform worse on the circle than on the interval or disk. This can be explained by comparing the cost of a bad decision on the circle to the cost of a bad decision on the interval or disk. Note that on the circle, every customer is at the farthest possible location from the depot (at a distance of 1), however on the interval or on the disk, that is not the case. Therefore, a wrong decision such as serving a customer alone instead of combining it with others is penalized with the maximum penalty on the circle. And finally, it should be mentioned that the performance of dynamic algorithms is expected to decrease if customer locations are not uniformly distributed in a region. This suggests that the value of stochastic information is bigger for problem settings with highly variable customer locations and consequently incorporating this information into decision making becomes more important.

Table 19: Performance of SMART(p) for multi-customer instances

Number	Density	Var	LB	UB	DDD	SMART(1.5)	SMART(2)	IDID
1	High	High	20	50	436.457	327.119	302.737	302.807
2	High	High	20	50	434.606	315.66	301.281	301.84
3	High	High	20	50	420.587	310.466	298.881	298.464
4	High	Medium	30	40	425.305	296.512	295.642	297.113
5	High	Medium	30	40	423.831	297.441	296.935	299.064
6	High	Medium	30	40	429.741	302.349	296.113	297.241
7	High	Low	35	35	430.897	315.335	294.586	296.32
8	High	Low	35	35	424.679	294.496	294.632	294.194
9	High	Low	35	35	423.145	296.701	296.033	296.975
10	Medium	High	0	30	267.318	195.071	190.89	194.929
11	Medium	High	0	30	250.887	181.098	178.612	177.409
12	Medium	High	0	30	263.429	194.461	205.061	197.643
13	Medium	Medium	10	20	299.454	218.122	213.683	210.434
14	Medium	Medium	10	20	288.203	200.486	201.292	201.18
15	Medium	Medium	10	20	291.477	202.794	197.9	199.077
16	Medium	Low	15	15	292.737	203.692	203.901	204.304
17	Medium	Low	15	15	296.915	202.355	197.144	197.29
18	Medium	Low	15	15	300.919	210.003	203.182	203.603
19	Low	High	0	15	183.619	140.957	136.022	135.575
20	Low	High	0	15	205.782	150.277	149.853	147.784
21	Low	High	0	15	230.046	173.896	169.259	163.452
22	Low	Medium	5	10	204.015	140.142	144.284	142.723
23	Low	Medium	5	10	209.702	151.294	154.639	150.765
24	Low	Medium	5	10	195.561	149.014	140.252	143.847
25	Low	Low	7	8	204.296	154.548	154.496	145.851
26	Low	Low	7	8	214.577	152.257	146.946	146.946
27	Low	Low	7	8	218.973	154.771	152.566	152.566
28	Very Low	High	0	6	121.892	99.949	98.8956	95.6444
29	Very Low	High	0	6	124.257	90.3127	88.9402	88.9402
30	Very Low	High	0	6	133.448	102.573	102.975	105.445
31	Very Low	Medium	2	4	137.47	104.834	103.597	102.382
32	Very Low	Medium	2	4	136.768	100.179	100.589	100.873
33	Very Low	Medium	2	4	132.59	99.8351	100.858	99.9977
34	Very Low	Low	3	3	137.64	103.584	101.133	99.2575
35	Very Low	Low	3	3	130.468	99.4688	96.4322	95.5535
36	Very Low	Low	3	3	140.151	102.455	99.2105	99.2105
				AVE=	**262.829**	**189.847**	**186.374**	**185.742**

Table 20: Performance of different implementations of SMART(p)

p	**Implementation 2**	**Implementation 1**
1.1	212.795	214.603
1.5	189.847	192.755
2	186.374	187.158
3	186.095	186.237
10000	185.742	186.039

Table 21: Performance of the Sampling-based Policy

Seed	LB	UB	IDID	Sampling n=50, 25 >=	Sampling n=50, 15 >=	Sampling n=50, Average Cost
1	20	50	304.365	323.24	305.047	402.439
2	20	50	302.472	308.813	302.592	393.057
3	20	50	298.642	317.222	313.894	391.345
4	30	40	296.544	321.5	302.67	380.506
5	30	40	297.330	315.774	294.542	385.369
6	30	40	298.956	305.598	300.84	391.26
7	35	35	295.962	298.396	304.437	384.922
8	35	35	293.610	317.572	292.108	384.513
9	35	35	297.980	306.172	299.755	385.42
10	0	30	195.010	200.305	190.598	236.737
11	0	30	177.490	179.944	178.451	222.439
12	0	30	197.938	204.415	200.582	234.038
13	10	20	209.871	217.662	211.568	260.343
14	10	20	200.711	203.651	204.644	243.461
15	10	20	199.555	207.795	203.532	249.132
16	15	15	203.833	208.933	203.172	249.258
17	15	15	198.179	208.103	198.156	252.133
18	15	15	206.987	212.642	207.142	258.022
19	0	15	134.281	133.243	131.885	153.577
20	0	15	147.593	150.382	150.8	174.557
21	0	15	164.539	168.264	165.78	200.136
22	5	10	142.389	141.627	142.214	171.274
23	5	10	150.696	154.509	146.869	177.16
24	5	10	144.402	140.546	137.138	167.72
25	7	8	145.915	144.811	146.972	177.321
26	7	8	147.359	148.939	143.693	172.832
27	7	8	153.558	158.152	152.815	183.413
28	0	6	95.978	97.2113	93.3789	101.437
29	0	6	90.2577	88.984	88.984	97.2041
30	0	6	106.156	101.9	101.085	110.323
31	2	4	103.137	101.221	99.9041	112.468
32	2	4	100.985	103.441	101.382	110.032
33	2	4	100.208	99.7038	99.2286	109.031
34	3	3	99.5656	100.253	99.3376	117.238
35	3	3	95.5535	93.7002	92.1101	106.366
36	3	3	99.3783	103.138	100.589	114.687
		Ave=	**186.039**	**191.327**	**186.330**	**229.477**

Table 22: Performance of Sampling Approach for Fewer Customers on the Plane

Number	Density	LB	UB	Offline Optimal	Sampling n=10, 5 >=	IDID
1	Very Low	1	1	47.6551	49.9528	48.6342
2	Very Low	1	1	49.2679	50.5463	52.6168
3	Very Low	1	1	48.3994	50.0872	52.7193
4	Very Low	1	1	49.7098	52.7141	55.153
5	Very Low	1	1	44.9093	45.7346	51.0661
6	Very Low	1	2	60.8496	63.5498	65.9601
7	Very Low	1	2	67.1205	68.8936	71.9748
8	Very Low	1	2	65.1473	67.4116	71.4964
9	Very Low	1	2	65.5246	69.789	70.8242
10	Very Low	1	2	62.0559	65.7772	64.9265
11	Very Low	1	3	71.8232	76.5536	72.4716
12	Very Low	1	3	74.56	78.7105	79.7284
13	Very Low	1	3	85.5473	90.0528	89.4842
14	Very Low	1	3	74.3627	78.0933	81.6918
15	Very Low	1	3	75.4064	78.4797	81.3962
16	Very Low	1	4	79.9734	89.0056	82.6082
17	Very Low	1	4	85.7658	91.7394	91.9668
18	Very Low	1	4	80.9584	91.3817	84.4079
19	Very Low	1	4	76.0222	80.0577	80.7447
20	Very Low	1	4	89.6181	95.89	95.9753
			Average=	67.733845	71.721025	72.292325

Table 23: Non-Uniform Settings

Setting	Number of Squares	Half Widths	Percentages
1	2	0.1,0.1	50,50
2	2	0.1,0.2	50,50
3	2	0.2,0.2	50,50
4	2	0.1,0.1	66,34
5	2	0.1,0.2	66,34
6	2	0.2,0.2	66,34
7	3	0.1,0.1,0.1	33,33,34
8	3	0.1,0.1,0.2	33,33,34
9	3	0.1,0.2,0.2	33,33,34
10	3	0.2,0.2,0.2	33,33,34
11	3	0.1,0.1,0.1	60,20,20
12	3	0.1,0.1,0.2	60,20,20
13	3	0.1,0.2,0.2	60,20,20
14	3	0.2,0.2,0.2	60,20,20
15	5	0.1,0.1,0.1,0.1,0.1	20,20,20,20,20
16	5	0.1,0.1,0.1,0.1,0.1	56,11,11,11,11

Table 24: Non-Uniform Customers Comparison Btw IDID and Sampling

Setting	Number of Squares	Half Widths	Percentages	IDID	Sampling n=50, 25 >=	Percent Improvement
1	2	0.1,0.1	50,50	100.245	98.555	1.685
2	2	0.1,0.2	50,50	110.700	108.598	1.899
3	2	0.2,0.2	50,50	119.917	117.763	1.796
4	2	0.1,0.1	66,34	99.605	98.101	1.510
5	2	0.1,0.2	66,34	110.034	105.556	4.070
6	2	0.2,0.2	66,34	118.712	116.056	2.238
7	3	0.1,0.1,0.1	33,33,34	112.121	110.436	1.503
8	3	0.1,0.1,0.2	33,33,34	121.317	119.212	1.735
9	3	0.1,0.2,0.2	33,33,34	127.198	124.859	1.839
10	3	0.2,0.2,0.2	33,33,34	132.818	129.808	2.266
11	3	0.1,0.1,0.1	60,20,20	112.490	112.631	-0.125
12	3	0.1,0.1,0.2	60,20,20	120.400	117.906	2.071
13	3	0.1,0.2,0.2	60,20,20	126.052	121.348	3.732
14	3	0.2,0.2,0.2	60,20,20	131.695	131.322	0.283
15	5	0.1,0.1,0.1,0.1,0.1	20,20,20,20,20	129.061	129.849	-0.610
16	5	0.1,0.1,0.1,0.1,0.1	56,11,11,11,11	122.509	123.073	-0.460
			Average=	**118.430**	**116.567**	**1.590**

APPENDIX A

PROOFS OF CHAPTER 3

R0. For any customer set $\{v_1, v_2, ..., v_x\}$ with $d_{v_i} \geq d_{v_j}$ for $i \leq j$, then we have: $d_{v_1} \geq p_{v_2} d_{v_2} + p_{v_3} d_{v_3}(1-p_{v_2}) + ... + p_{v_x} d_{v_x} \prod_{i=v_2}^{v_{x-1}}(1-p_i)$. This holds if;

$d_{v_1} \geq p_{v_2} d_{v_1} + p_{v_3} d_{v_1}(1-p_{v_2}) + ... + p_{v_x} d_{v_1} \prod_{i=v_2}^{v_{x-1}}(1-p_i)$ or $1 \geq [1 - \prod_{i=v_2}^{v_x}(1-p_i)]$.

R1. Let $B' = \{\{v_1, v_2, ..., v_x\}$ with $d_{v_i} \geq d_{v_j}$ for $i \leq j$. Letting $\alpha = \prod_{j \subseteq A}(1-p_j)$;

$M = d_k * \alpha - p_{v_1} d_{v_1} \alpha - p_{v_2} d_{v_2} \alpha (1-p_{v_1}) - ... - p_{v_x} d_{v_x} \alpha \prod_{i=v_1}^{v_{x-1}}(1-p_i) = \alpha[d_k - p_{v_1} d_{v_1} - p_{v_2} d_{v_2}(1-p_{v_1}) - ... - p_{v_x} d_{v_x} \prod_{i=v_1}^{v_{x-1}}(1-p_i)]$. The term in parenthesis is nonnegative (from R0) and constant for a given B', so to minimize M, α should be made as small as possible, hence leading R1.

R2. Let A' be given. Let $B_1 = \{k+1, k+2, ..., y-1, y, v_1, v_2, ..., v_x\}$ with $v_1 > y+1$ and $d_{v_i} \geq d_{v_j}$ for $i \leq j$ so that selection B_1 obeys R2 until customer y, but customer $y+1$ is not selected. Let $B_2 = \{k+1, k+2, ..., y-1, y, y+1, v_2, ..., v_x\}$. We show that M is decreased from B_1 to B_2. Letting $\alpha = \prod_{j \in A'}(1-p_j)$ and $\beta = \prod_{j=k+1}^{y}(1-p_j)$;

$M(B_1) = \alpha[d_k - p_{k+1} d_{k+1} - ... - p_y d_y \prod_{i=k+1}^{y-1}(1-p_i)] - \alpha\beta[p_{v_1} d_{v_1} - p_{v_2} d_{v_2}(1-p_{v_1}) - ... - p_{v_x} d_{v_x} \prod_{i=v_1}^{v_{x-1}}(1-p_i)]$.

$M(B_2) = \alpha[d_k - p_{k+1} d_{k+1} - ... - p_y d_y \prod_{i=k+1}^{y-1}(1-p_i)] - \alpha\beta[p_{y+1} d_{y+1} - p_{v_2} d_{v_2}(1-p_{y+1}) - ... - p_{v_x} d_{v_x}(1-p_{y+1}) \prod_{i=v_2}^{v_{x-1}}(1-p_i)]$. $M(B_2) \leq M(B_1)$ iff:

$[p_{y+1} d_{y+1} - p_{v_2} d_{v_2}(1-p_{y+1}) - ... - p_{v_x} d_{v_x}(1-p_{y+1}) \prod_{i=v_2}^{v_{x-1}}(1-p_i)] \geq [p_{v_1} d_{v_1} - p_{v_2} d_{v_2}(1-p_{v_1}) - ... - p_{v_x} d_{v_x} \prod_{i=v_1}^{v_{x-1}}(1-p_i)]$. Replacing d_{v_1} by d_{y+1}, we seek if:

$d_{y+1}[p_{y+1} - p_{v_1}] \geq [p_{v_1} - p_{y+1}][p_{v_2} d_{v_2} + ... + p_{v_x} d_{v_x} \prod_{i=v_2}^{v_{x-1}} (1-p_i)]$. Since $p_{y+1} \geq p_{v_1}$, this inequality holds.

R3. Let $A_1 = \{1, 2, ..., x\}$ and $B_1 = \{k+1, k+2, ..., y\}$ as in Figure 8 with $x < k-1$ so that there are some customers unselected among set A. Let $A_2 = \{1, 2, ..., x, x+1\}$ and $B_2 = \{k+1, k+2, ..., y-1\}$. We show that M is decreased from (A_1, B_1) to (A_2, B_2). Letting $\alpha = \prod_{i=1}^{x}(1-p_i)$;

$M(A_1, B_1) = \alpha[d_k - p_{k+1} d_{k+1} - ... - p_y d_y \prod_{i=k+1}^{y-1} (1-p_i)$.

$M(A_2, B_2) = \alpha(1 - p_{x+1})[d_k - p_{k+1} d_{k+1} - ... - p_{y-1} d_{y-1} \prod_{i=k+1}^{y-2} (1-p_i)$.

Letting $X = d_k - p_{k+1} d_{k+1} - ... - p_{y-1} d_{y-1} \prod_{i=k+1}^{y-2}(1-p_i)$, we rewrite:

$M(A_1, B_1) - M(A_2, B_2) = \alpha[X - p_y d_y \prod_{i=k+1}^{y-1} (1-p_i)] - \alpha(1-p_{x+1})X \geq 0$ if; $[X - p_y d_y \prod_{i=k+1}^{y-1}(1-p_i)] \geq (1-p_{x+1})X$, or if $X p_{x+1} \geq p_y d_y \prod_{i=k+1}^{y-1} (1-p_i)$. Since $p_{x+1} \geq p_y$, we seek:

$X = d_k - p_{k+1} d_{k+1} - ... - p_{y-1} d_{y-1} \prod_{i=k+1}^{y-2}(1-p_i) \geq d_y \prod_{i=k+1}^{y-1}(1-p_i)$. Replacing all d_i for $k+1 \leq i \leq y$ by d_{k+1}, we seek if:

$d_k \geq d_{k+1}[1 - \prod_{i=k+1}^{y-1} (1-p_i)] + d_{k+1} \prod_{i=k+1}^{y-1} (1-p_i) = d_{k+1}$, which is true.

S1. Let $A' = \{\{v_1, v_2, ..., v_x\}$ with $d_{v_i} \geq d_{v_j}$ for $i \leq j$. Letting $\alpha = \prod_{j \subseteq B} (1-p_j)$;

$H = \alpha[d_{a+k} - p_{v_1} d_{v_1} - ... - p_{v_x} d_{v_x} \prod_{i=v_1}^{v_{x-1}} (1-p_i)]$. The term in parenthesis is non-negative (from R0) and constant for a given A'. To maximize M, α should be made as large as possible, hence leading S1.

S2. Let B' be given. Let $A_1 = \{v_1, v_2, ..., v_x, y, y+2, y+3, ..., a+b\}$ with $d_{v_x} \geq d_y$ and $d_{v_i} \geq d_{v_j}$ for $i \leq j$ so that selection A_1 obeys S2 until customer $y+2$, but customer $y+1$ is not selected. Let $A_2 = \{v_1, v_2, ..., v_x, y+1, y+2, y+3, ..., a+b\}$. We show that H increases from A_1 to A_2. Letting $\alpha = \prod_{j \in B'} (1-p_j)$ and $\beta = \prod_{i=v_1}^{v_x} (1-p_i)$;

$H(A_1) = \alpha[d_{a+k} - p_{v_1} d_{v_1} - .. - p_{v_x} d_{v_x} \prod_{i=v_1}^{v_{x-1}} (1-p_i)] - \alpha\beta[p_y d_y + p_{y+2} d_{y+2}(1-p_y) + ... +$

$p_{a+b}d_{a+b}(1-p_y)\prod_{i=y+2}^{a+b-1}(1-p_i)]$

$H(A_2) = \alpha[d_{a+k} - p_{v_1}d_{v_1} - .. - p_{v_x}d_{v_x}\prod_{i=v_1}^{v_x-1}(1-p_i)] - \alpha\beta[p_{y+1}d_{y+1} + p_{y+2}d_{y+2}(1-p_{y+1}) +$
$... + p_{a+b}d_{a+b}(1-p_{y+1})\prod_{i=y+2}^{a+b-1}(1-p_i)]$

$H(A_2) \geq H(A_1)$ if $[p_{y+1}d_{y+1} + p_{y+2}d_{y+2}(1-p_{y+1}) + ... + p_{a+b}d_{a+b}(1-p_{y+1})\prod_{i=y+2}^{a+b-1}(1-p_i)] \leq [p_y d_y + p_{y+2}d_{y+2}(1-p_y) + ... + p_{a+b}d_{a+b}(1-p_y)\prod_{i=y+2}^{a+b-1}(1-p_i)]$. Replacing d_{y+1} by d_y and denoting $Z = p_{y+2}d_{y+2} + ... + p_{a+b}d_{a+b}\prod_{i=y+2}^{a+b-1}(1-p_i)$; we check if $d_y(p_y - p_{y+1}) \geq Z(p_y - p_{y+1})$, which is true since $d_y \geq Z$ by R0.

S3. Let $A_1 = \{x+1, x+2, ..., a+b\}$ and $B_1 = \{y-1, y, ..., a+k-1\}$ as in Figure 10 with $x > a+k$ so that there is at least one unselected customer among set A. Let $A_2 = \{x, x+1, x+2, ..., a+b\}$ and $B_2 = \{y, ..., a+k-1\}$. We show that H is increased from (A_1, B_1) to (A_2, B_2).

Letting $\beta_1 = \prod_{i=y-1}^{a+k-1}(1-p_i)$ and $\beta_2 = \prod_{i=y}^{a+k-1}(1-p_i)$;

$H(A_1, B_1) = \beta_1[d_{a+k} - p_{x+1}d_{x+1} - ... - p_{a+b}d_{a+b}\prod_{i=x+1}^{a+b-1}(1-p_i)]$

$H(A_2, B_2) = \beta_2[d_{a+k} - p_x d_x - p_{x+1}d_{x+1}(1-p_x) - ... - p_{a+b}d_{a+b}\prod_{i=x}^{a+b-1}(1-p_i)]$

Denoting $X = p_{x+1}d_{x+1} + p_{x+2}d_{x+2}(1-p_{x+1}) + ... + p_{a+b}d_{a+b}\prod_{i=x+1}^{a+b-1}(1-p_i)$, we have:

$H(A_2, B_2) - H(A_1, B_1) = d_{a+k}[\beta_2 - \beta_1] - \beta_2 p_x d_x - X[\beta_2(1-p_x) - \beta_1]$. Since $\beta_1 = \beta_2(1-p_{y-1})$, using $\beta_2 - \beta_1 = \beta_2 p_{y-1}$ and $[\beta_2(1-p_x) - \beta_1] = \beta_2(p_{y-1} - p_x)$; we seek if:

$d_{a+k}[\beta_2 p_{y-1}] \geq \beta_2 p_x d_x + X[\beta_2(p_{y-1} - p_x)]$. Canceling β_2 and replacing d_{a+k} by d_x; we seek if:

$d_x p_{y-1} \geq p_x d_x + X[(p_{y-1} - p_x)]$, which is true since $p_{y-1}[d_x - X] \geq p_x[d_x - X]$ by RO ($d_x \geq X$).

Expressions for Expected Total Costs

Let X_1, X_2, X_3 be the number of arriving customers among the sets of customers $\{1,2,3,4\}$, $\{5,6\}$, and $\{7\}$ respectively in Solution 1 in Figure 24. For $p_i = p$ for all i, $X_1 = Binomial(4,p)$, $X_2 = Binomial(2,p)$, and $X_3 = Binomial(1,p)$. For notational convenience, let $p_i^j = Prob\{X_i = j\}$ and $E[r_k]$ show the cost component for route k.

$E[Solution1] = E[r_1] + E[r_2] + E[r_3] + E[r_4]$.

$E[r_1] = \sum_{j=0}^{4} p_1^j * E[r_1|X_1 = j] = (p_1^0 + p_1^1 + p_1^2) * 0 + p_1^3(\frac{3}{4}d_1 + \frac{1}{4}d_2) + p_1^4(d_1)$. As you can see, we condition on the number of arriving customers X_1. If X_1 is zero, one, or two, route 1 is not needed for feasibility so they are served by route 2. If $X_1 = 3$, then route 1 is needed for feasibility, so is used to serve the furthest two of the arriving customers. Given that $X_1 = 3$, with probability $\frac{3}{4}$ the furthest customer will be customer 1 and with probability $\frac{1}{4}$ the furthest customer will be customer 2. If $X_1 = 4$, then route 1 will serve the furthest two of them and incur a cost of d_1. Expected costs of other routes are written by conditioning in a similar fashion.

$E[r_2] = \sum_{j=0}^{4} p_1^j * E[r_2|X_1 = j] = p_1^0 * 0 + p_1^1[\frac{1}{4}(d_1 + d_2 + d_3 + d_4)] + p_1^2[\frac{1}{2}d_1 + \frac{1}{3}d_2 + \frac{1}{6}d_3] + p_1^3[\frac{1}{4}d_3 + \frac{3}{4}d_4] + p_1^4 d_3$.

$E[r_3] = \sum_{j=0}^{2} p_2^j * E[r_3|X_2 = j] = p_2^0 * 0 + p_2^1[p_1^0(\frac{1}{2}d_5 + \frac{1}{2}d_6) + p_1^1 * 0 + p_1^2(\frac{1}{2}d_5 + \frac{1}{2}d_6) + p_1^3 * 0 + p_1^4(\frac{1}{2}d_5 + \frac{1}{2}d_6)] + p_2^2[p_1^0 * d_5 + p_1^1 d_6 + p_1^2 d_5 + p_1^3 d_6 + p_1^4 d_5]$.

$E[r_4] = \sum_{j=0}^{1} p_3^j * E[r_4|X_3 = j] = p_3^0 * 0 + p_3^1[p_2^0 d_7 + p_2^1(p_1^0 * 0 + p_1^1 d_7 + p_1^2 * 0 + p_1^3 d_7 + p_1^4 * 0) + p_2^2(p_1^0 d_7 + p_1^1 * 0 + p_1^2 d_7 + p_1^3 * 0 + p_1^4 d_7)]$.

Expected total cost of Solution 2 can be written too. However care must be taken to note that if a route is not needed for feasibility, then we may still use it and have a better solution. Suppose Y_1 and Y_2 be the number of arriving customers among the sets of customers $\{1, 2, 3, 4\}$ and $\{5, 6, 7\}$, respectively in Solution 2 in Figure 24. For $p_i = p$ for all i, $Y_1 = Binomial(4, p)$, and $Y_2 = Binomial(3, p)$. Let $p_i^j = Prob\{Y_i = j\}$.

$E[Solution2] = E[r_1] + E[r_2] + E[r_3] + E[r_4]$.

$E[r_1] = \sum_{j=0}^{4} p_1^j * E[r_1|Y_1 = j] = (p_1^0 + p_1^1) * 0 + + p_1^2(\frac{1}{2}d_1 + \frac{1}{3}d_2 + \frac{1}{6}d_3) + p_1^3(\frac{3}{4}d_1 + \frac{1}{4}d_2) + p_1^4(d_1)$.
Note that if Y_1 equals zero or one, then optimal operational level decision is not to use route 1. If $Y_1 = 2$, although route 1 is not needed for feasibility, optimal decision is to use route 1 to serve these two customers, because route 2 can be used to combine customers among Y_2.

$E[r_2] = \sum_{j=0}^{4} p_1^j * E[r_2|Y_1 = j] = p_1^0[p_2^0 * 0 + p_2^1(\frac{1}{3}(d_5 + d_6 + d_7)) + p_2^2(\frac{2}{3}d_5 + \frac{1}{3}d_6) + p_2^3 d_5] +$

$p_1^1[\frac{1}{4}(d_1+d_2+d_3+d_4)]+p_1^2[p_2^0*0+p_2^1(\frac{1}{3}(d_5+d_6+d_7))+p_2^2(\frac{2}{3}d_5+\frac{1}{3}d_6)+p_2^3d_5]+p_1^3[\frac{1}{4}d_3+\frac{3}{4}d_4]+p_1^4d_3.$

$E[r_3]=\sum_{j=0}^{3}p_2^j*E[r_3|Y_2=j]=p_2^0*0+p_2^1[(p_1^0+p_1^1+p_1^2+p_1^3)*0+p_1^4(\frac{1}{3}d_5+\frac{1}{3}d_6+\frac{1}{3}*0)]+p_2^2[p_1^0*0+p_1^1(\frac{1}{3}d_6+\frac{1}{3}*0+\frac{1}{3}*0)+p_1^2*0+p_1^3(\frac{1}{3}d_6+\frac{1}{3}*0+\frac{1}{3}*0)+p_1^4(\frac{2}{3}d_5+\frac{1}{3}d_6)]+p_2^3[p_1^0*0+p_1^1d_6+p_1^2*0+p_1^3d_6+p_1^4d_5].$

$E[r_4]=\sum_{j=0}^{3}p_2^j*E[r_4|Y_2=j]=p_2^0*0+p_2^1[(p_1^0+p_1^1+p_1^2+p_1^3)*0+p_1^4(\frac{2}{3}*0+\frac{1}{3}d_7)]+p_2^2[p_1^0*0+p_1^1\frac{2}{3}d_7+p_1^2*0+p_1^3\frac{2}{3}d_7+p_1^4\frac{2}{3}d_7]+p_2^3[p_1^0d_7+p_1^1d_7+p_1^2d_7+p_1^3d_7+p_1^4d_7].$

APPENDIX B

PROOFS OF CHAPTER 4

B.0.1 Proof of Proposition 6

We first establish some intermediate results before we prove Proposition 6.

Lemma 16 *Consider a set* $\mathbb{X}$ *and two functions* $f_1, f_2 : \mathbb{X} \mapsto \mathbb{R}$*. Let* $f_\wedge, f_\vee : \mathbb{X} \mapsto \mathbb{R}$ *be given by* $f_\wedge(x) := \min\{f_1(x), f_2(x)\}$*,* $f_\vee(x) := \max\{f_1(x), f_2(x)\}$*. Then, for any* $x_1, x_2 \in \mathbb{X}$*,*

$$f_\wedge(x_2) - f_\wedge(x_1) \leq \max\{f_1(x_2) - f_1(x_1), f_2(x_2) - f_2(x_1)\} \tag{16}$$

$$f_\vee(x_2) - f_\vee(x_1) \leq \max\{f_1(x_2) - f_1(x_1), f_2(x_2) - f_2(x_1)\} \tag{17}$$

Thus, for any $x_1, x_2 \in \mathbb{X}$*,*

$$\begin{aligned} |f_\wedge(x_2) - f_\wedge(x_1)| &\leq \max\{|f_1(x_2) - f_1(x_1)|, |f_2(x_2) - f_2(x_1)|\} \\ |f_\vee(x_2) - f_\vee(x_1)| &\leq \max\{|f_1(x_2) - f_1(x_1)|, |f_2(x_2) - f_2(x_1)|\} \end{aligned}$$

Consider a function $d : \mathbb{X} \times \mathbb{X} \mapsto [0, \infty)$*, and suppose that there are constants* L_i*,* $i = 1, 2$*, such that for any* $x_1, x_2 \in \mathbb{X}$*,*

$$|f_i(x_2) - f_i(x_1)| \leq L_i d(x_1, x_2) \qquad i = 1, 2$$

Then

$$\begin{aligned} |f_\wedge(x_2) - f_\wedge(x_1)| &\leq \max\{L_1, L_2\} d(x_1, x_2) \\ |f_\vee(x_2) - f_\vee(x_1)| &\leq \max\{L_1, L_2\} d(x_1, x_2) \end{aligned}$$

Thus, if each f_i *is Lipschitz continuous with Lipschitz constant* L_i*, then* $f_\wedge, f_\vee$ *are Lipschitz continuous with Lipschitz constant* $\max\{L_1, L_2\}$*.*

Proof. Without loss of generality, suppose that $f_1(x_1) \le f_2(x_1)$. Then

$$\begin{aligned}
\min\{f_1(x_2), f_2(x_2)\} - \min\{f_1(x_1), f_2(x_1)\} &= \min\{f_1(x_2), f_2(x_2)\} - f_1(x_1) \\
&\le f_1(x_2) - f_1(x_1) \\
&\le \max\{f_1(x_2) - f_1(x_1), f_2(x_2) - f_2(x_1)\}
\end{aligned}$$

The result (17) for $f_\vee$ follows in a similar way.

Next, without loss of generality, suppose that $f_\wedge(x_1) \le f_\wedge(x_2)$. Then

$$\begin{aligned}
|f_\wedge(x_2) - f_\wedge(x_1)| &= \min\{f_1(x_2), f_2(x_2)\} - \min\{f_1(x_1), f_2(x_1)\} \\
&\le \max\{f_1(x_2) - f_1(x_1), f_2(x_2) - f_2(x_1)\} \\
&\le \max\{L_1 d(x_1, x_2), L_2 d(x_1, x_2)\} \\
&= \max\{L_1, L_2\} d(x_1, x_2)
\end{aligned}$$

The result for $f_\vee$ follows in a similar way. $\square$

Lemma 17 *Consider a set $\mathbb{X}$, a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, a function $f_\times : \mathbb{X} \times \Omega \mapsto \mathbb{R}$, and a function $d : \mathbb{X} \times \mathbb{X} \mapsto [0, \infty)$. Suppose that $f_\times(x, \cdot)$ is integrable for each $x \in \mathbb{X}$, and there exists a measurable function $L : \Omega \mapsto \mathbb{R}$ such that for each $x_1, x_2 \in \mathbb{X}$ and each $\omega \in \Omega$,*

$$|f_\times(x_2, \omega) - f_\times(x_1, \omega)| \le L(\omega) d(x_1, x_2)$$

Then

$$\left| \int_\Omega f_\times(x_2, \omega)\, d\mathbb{P}(\omega) - \int_\Omega f_\times(x_1, \omega)\, d\mathbb{P}(\omega) \right| \le \int_\Omega L(\omega)\, d\mathbb{P}(\omega)\ d(x_1, x_2)$$

Thus, if $f_\times$ is Lipschitz continuous in $\mathbb{X}$ with Lipschitz constant L for each $\omega \in \Omega$, then $\int_\Omega f_\times(x, \omega) d\mathbb{P}(\omega)$ is Lipschitz continuous (in $\mathbb{X}$) with Lipschitz constant L.

Proof.

$$\begin{aligned}
\left| \int_\Omega f_\times(x_2, \omega)\, \mathbb{P}(d\omega) - \int_\Omega f_\times(x_1, \omega)\, \mathbb{P}(d\omega) \right| &\le \int_\Omega |f_\times(x_2, \omega) - f_\times(x_1, \omega)|\, \mathbb{P}(d\omega) \\
&\le \int_\Omega L(\omega) d(x_1, x_2)\, \mathbb{P}(d\omega)
\end{aligned}$$

$\square$

Lemma 18 *Consider any function $V : [0,1] \mapsto \mathbb{R}$ and a probability distribution F on $[0,1]$. For any $0 \le x_1 \le x_2 \le 1$ it holds that*

$$\begin{aligned} 0 \quad \le \quad & \int_{[0,1]} \min\left\{x_2 + \alpha V(\xi),\ \max\{x_2,\xi\} + \alpha V(0)\right\}\ dF(\xi) \\ & - \int_{[0,1]} \min\left\{x_1 + \alpha V(\xi),\ \max\{x_1,\xi\} + \alpha V(0)\right\}\ dF(\xi) \quad \le \quad x_2 - x_1 \end{aligned}$$

Proof. It follows from $x_1 \le x_2$ that $\min\left\{x_1 + \alpha V(\xi), \max\{x_1,\xi\} + \alpha V(0)\right\} \le$
$\min\left\{x_2 + \alpha V(\xi), \max\{x_2,\xi\} + \alpha V(0)\right\}$ for all ξ, and thus
$\int_{[0,1]} \min\left\{x_1 + \alpha V(\xi), \max\{x_1,\xi\} + \alpha V(0)\right\} dF(\xi) \le$
$\int_{[0,1]} \min\left\{x_2 + \alpha V(\xi), \max\{x_2,\xi\} + \alpha V(0)\right\} dF(\xi)$.
Note that for each ξ,

$$\begin{aligned} [x_2 + \alpha V(\xi)] - [x_1 + \alpha V(\xi)] \quad &= \quad x_2 - x_1 \\ [\max\{x_2,\xi\} + \alpha V(0)] - [\max\{x_1,\xi\} + \alpha V(0)] \quad &\le \quad x_2 - x_1 \end{aligned}$$

Thus it follows from Lemma 16 that for each ξ,

$$\begin{aligned} & \min\left\{x_2 + \alpha V(\xi),\ \max\{x_2,\xi\} + \alpha V(0)\right\} - \min\left\{x_1 + \alpha V(\xi),\ \max\{x_1,\xi\} + \alpha V(0)\right\} \\ \le \quad & \max\{[x_2 + \alpha V(\xi)] - [x_1 + \alpha V(\xi)], [\max\{x_2,\xi\} + \alpha V(0)] - [\max\{x_1,\xi\} + \alpha V(0)]\} \\ = \quad & x_2 - x_1 \end{aligned}$$

Next it follows from Lemma 17 that

$$\begin{aligned} & \int_{[0,1]} \min\left\{x_2 + \alpha V(\xi),\ \max\{x_2,\xi\} + \alpha V(0)\right\}\ dF(\xi) \\ & - \int_{[0,1]} \min\left\{x_1 + \alpha V(\xi),\ \max\{x_1,\xi\} + \alpha V(0)\right\}\ dF(\xi) \quad \le \quad x_2 - x_1 \end{aligned}$$

□

Proposition 6 follows from the optimality equation (6) and Lemma 18.

B.0.2 Proofs for Section 4.2.1.2

Let $\mathbb{X}$ denote the state space, and let $\mathcal{V}$ denote the set of bounded value functions $V : \mathbb{X} \mapsto \mathbb{R}$. Let $\|V\| := \sup_{x \in \mathbb{X}} |V(x)|$. Consider the mapping $T : \mathcal{V} \mapsto \mathcal{V}$ given by

$$T(V)(x) \quad := \quad \int_{\Xi} \inf_u \{c(x,\xi,u) + \alpha V(f(x,\xi,u))\}\ d\mathbb{P}(\xi)$$

Consider any $V_1, V_2 \in \mathcal{V}$ and any $x \in \mathbb{X}$. Then

$$
\begin{aligned}
&|T(V_2)(x) - T(V_1)(x)| = \left| \int_\Xi \inf_u \{c(x,\xi,u) + \alpha V_2(f(x,\xi,u))\} \, d\mathbb{P}(\xi) \right. \\
&\left. - \int_\Xi \inf_u \{c(x,\xi,u) + \alpha V_1(f(x,\xi,u))\} \, d\mathbb{P}(\xi) \right| \\
&\leq \int_\Xi \left| \inf_u \{c(x,\xi,u) + \alpha V_2(f(x,\xi,u))\} - \inf_u \{c(x,\xi,u) + \alpha V_1(f(x,\xi,u))\} \right| \, d\mathbb{P}(\xi) \\
&\leq \int_\Xi \sup_u |\{c(x,\xi,u) + \alpha V_2(f(x,\xi,u))\} - \{c(x,\xi,u) + \alpha V_1(f(x,\xi,u))\}| \, d\mathbb{P}(\xi) \\
&\leq \int_\Xi \sup_{y\in\mathbb{X}} |\alpha V_2(y) - \alpha V_1(y)| \, d\mathbb{P}(y) \\
&= \int_\Xi \alpha \|V_2 - V_1\| \, d\mathbb{P}(\xi) \\
&= \alpha \|V_2 - V_1\|
\end{aligned}
$$

Thus

$$
\|T(V_2) - T(V_1)\| \quad := \quad \sup_{x\in\mathbb{X}} |T(V_2)(x) - T(V_1)(x)| \quad \leq \quad \alpha \|V_2 - V_1\| \tag{18}
$$

that is, T is a contraction mapping with contraction factor α and unique fixed point $V^\infty = T(V^\infty)$. For any $V \in \mathcal{V}$, let $T^0(V) := V$ and $T^{i+1}(V) := T(T^i(V))$ for $i = 0, 1, \ldots$. It is easily seen that T^i has contraction factor α^i. Then for any $V \in \mathcal{V}$, $\left\|V^\infty - T^i(V)\right\| \to 0$ as $i \to \infty$. Note that the results above hold irrespective of the functions f, g, or the measure $\mathbb{P}$.

Proof of Lemma 7. Note that

$$
V_2 - V_1 \quad = \quad \sum_{i=0}^{\infty} [T_2^{i+1}(V_1) - T_2^i(V_1)]
$$

Thus

$$\begin{aligned}
\|V_2 - V_1\| &= \left\| \sum_{i=0}^{\infty} [T_2^{i+1}(V_1) - T_2^i(V_1)] \right\| \\
&\leq \sum_{i=0}^{\infty} \left\| T_2^{i+1}(V_1) - T_2^i(V_1) \right\| \\
&= \sum_{i=0}^{\infty} \left\| T_2^i(T_2(V_1)) - T_2^i(T_1(V_1)) \right\| \\
&\leq \sum_{i=0}^{\infty} \alpha^i \left\| T_2(V_1) - T_1(V_1) \right\| \\
&\leq \sum_{i=0}^{\infty} \alpha^i \varepsilon \\
&= \frac{\varepsilon}{1-\alpha}
\end{aligned}$$

$\square$

Proof of Lemma 8. Note that

$$V^{\infty} - T(V) = \sum_{i=1}^{\infty} [T^{i+1}(V) - T^i(V)]$$

Thus

$$\begin{aligned}
\|V^{\infty} - T(V)\| &= \left\| \sum_{i=1}^{\infty} [T^{i+1}(V) - T^i(V)] \right\| \\
&\leq \sum_{i=1}^{\infty} \left\| T^{i+1}(V) - T^i(V) \right\| \\
&= \sum_{i=1}^{\infty} \left\| T^i(T(V)) - T^i(V) \right\| \\
&\leq \sum_{i=1}^{\infty} \alpha^i \left\| T(V) - V \right\| \\
&\leq \sum_{i=1}^{\infty} \alpha^i \vartheta \\
&= \frac{\alpha\vartheta}{1-\alpha}
\end{aligned}$$

$\square$

Proof of Proposition 9. The result follows from Lemma 7, Lemma 8, and the triangle inequality:

$$\left\|V^* - \hat{T}(V)\right\| \leq \left\|V^* - \hat{V}\right\| + \left\|\hat{V} - \hat{T}(V)\right\| \leq \frac{\varepsilon + \alpha\vartheta}{1-\alpha}$$

□

Proof of Proposition 10. For any $V \in \mathcal{V}$, let

$$T^{\hat{\pi}}(V)(x) := \int_{[0,1]} [g(x, y, \hat{\pi}(x, y)) + \alpha V(f(x, y, \hat{\pi}(x, y)))] \; dF(y)$$

Note that $T^{\hat{\pi}}(V^{\hat{\pi}}) = V^{\hat{\pi}}$. It follows from (18) that $T^{\hat{\pi}}$ is a contraction mapping with contraction factor α, and thus $V^{\hat{\pi}}$ is the unique fixed point of $T^{\hat{\pi}}$, and for any $V \in \mathcal{V}$, $(T^{\hat{\pi}})^i(V) \to V^{\hat{\pi}}$ as $i \to \infty$. It follows from the definition of $\hat{\pi}$ that $T^{\hat{\pi}}(\hat{T}^2(V)) = T^*(\hat{T}^2(V))$.

Consider

$$\left\|V^* - V^{\hat{\pi}}\right\| \leq \left\|V^* - \hat{T}^2(V)\right\| + \left\|\hat{T}^2(V) - V^{\hat{\pi}}\right\|$$

First, it follows as in Lemma 8 that

$$\begin{aligned}
\left\|\hat{V} - \hat{T}^2(V)\right\| &= \left\|\sum_{i=2}^{\infty}[\hat{T}^{i+1}(V) - \hat{T}^i(V)]\right\| \\
&\leq \sum_{i=2}^{\infty}\left\|\hat{T}^{i+1}(V) - \hat{T}^i(V)\right\| \\
&= \sum_{i=2}^{\infty}\left\|\hat{T}^i(\hat{T}(V)) - \hat{T}^i(V)\right\| \\
&\leq \sum_{i=2}^{\infty}\alpha^i\left\|\hat{T}(V) - V\right\| \\
&\leq \sum_{i=2}^{\infty}\alpha^i\vartheta \\
&= \frac{\alpha^2\vartheta}{1-\alpha}
\end{aligned}$$

and thus it follows from Lemma 7 that

$$\left\|V^* - \hat{T}^2(V)\right\| \leq \left\|V^* - \hat{V}\right\| + \left\|\hat{V} - \hat{T}^2(V)\right\| \leq \frac{\varepsilon + \alpha^2\vartheta}{1-\alpha}$$

Second,

$$
\begin{aligned}
\left\|\hat{T}^2(V) - V^{\hat{\pi}}\right\| &= \left\|\hat{T}^2(V) - T^{\hat{\pi}}(V^{\hat{\pi}})\right\| \\
&\leq \left\|\hat{T}^2(V) - T^{\hat{\pi}}(\hat{T}^2(V))\right\| + \left\|T^{\hat{\pi}}(\hat{T}^2(V)) - T^{\hat{\pi}}(V^{\hat{\pi}})\right\| \\
&\leq \left\|\hat{T}^2(V) - T^{\hat{\pi}}(\hat{T}^2(V))\right\| + \alpha \left\|\hat{T}^2(V) - V^{\hat{\pi}}\right\| \\
\Rightarrow \quad (1-\alpha)\left\|\hat{T}^2(V) - V^{\hat{\pi}}\right\| &\leq \left\|\hat{T}^2(V) - T^{\hat{\pi}}(\hat{T}^2(V))\right\| \\
&= \left\|\hat{T}^2(V) - T^{*}(\hat{T}^2(V))\right\| \\
&\leq \left\|\hat{T}^2(V) - \hat{T}(\hat{T}^2(V))\right\| + \left\|\hat{T}(\hat{T}^2(V)) - T^{*}(\hat{T}^2(V))\right\|
\end{aligned}
$$

Note that

$$
\begin{aligned}
\left\|\hat{T}^2(V) - \hat{T}(\hat{T}^2(V))\right\| &= \left\|\hat{T}(\hat{T}^2(V)) - \hat{T}(\hat{T}(V))\right\| \\
&\leq \alpha \left\|\hat{T}^2(V) - \hat{T}(V)\right\| \\
&= \alpha \left\|\hat{T}(\hat{T}(V)) - \hat{T}(V)\right\| \\
&\leq \alpha^2 \left\|\hat{T}(V) - V\right\| \\
&\leq \alpha^2 \vartheta
\end{aligned}
$$

Also, it follows from the second assumption that

$$
\left\|\hat{T}(\hat{T}^2(V)) - T^{*}(\hat{T}^2(V))\right\| \leq \varepsilon
$$

Putting these results together, it follows that

$$
\left\|V^{*} - V^{\hat{\pi}}\right\| \leq 2\frac{\varepsilon + \alpha^2\vartheta}{1-\alpha}
$$

□

Proof of equation (11). Suppose that both $V_i(x) = T_m(V_{i-1})(x)$ and $V_{i-1}(x)$ have been calculated at the points $x = 0, y_{m,1}, \ldots, y_{m,m}$. Next, suppose that

$$
\max_{x\in\{0,y_{m,1},\ldots,y_{m,m}\}} |V_i(x) - V_{i-1}(x)| \leq \vartheta
$$

Then, for any $x \in [0,1]$,

$$
\begin{aligned}
&|T_m(V_i)(x) - V_i(x)| = |T_m(V_i)(x) - T_m(V_{i-1})(x)| \\
&= \Bigg| \Bigg[\min\left\{x + \alpha V_i(y_{m,1}),\ \max\{x, y_{m,1}\} + \alpha V_i(0)\right\} F_m(y_{m,1}) \\
&\quad + \sum_{j=2}^{m} \min\left\{x + \alpha V_i(y_{m,j}),\ \max\{x, y_{m,j}\} + \alpha V_i(0)\right\} \left[F_m(y_{m,j}) - F_m(y_{m,j-1})\right] \Bigg] \\
&\quad - \Bigg[\min\left\{x + \alpha V_{i-1}(y_{m,1}),\ \max\{x, y_{m,1}\} + \alpha V_{i-1}(0)\right\} F_m(y_{m,1}) \\
&\quad + \sum_{j=2}^{m} \min\left\{x + \alpha V_{i-1}(y_{m,j}),\ \max\{x, y_{m,j}\} + \alpha V_{i-1}(0)\right\} \left[F_m(y_{m,j}) - F_m(y_{m,j-1})\right] \Bigg] \Bigg| \\
&\leq \left| \min\left\{x + \alpha V_i(y_{m,1}),\ \max\{x, y_{m,1}\} + \alpha V_i(0)\right\} F_m(y_{m,1}) \right. \\
&\left. - \min\left\{x + \alpha V_{i-1}(y_{m,1}),\ \max\{x, y_{m,1}\} + \alpha V_{i-1}(0)\right\} F_m(y_{m,1}) \right| \\
&+ \sum_{j=2}^{m} \left| \min\left\{x + \alpha V_i(y_{m,j}),\ \max\{x, y_{m,j}\} + \alpha V_i(0)\right\} \left[F_m(y_{m,j}) - F_m(y_{m,j-1})\right] \right. \\
&\qquad \left. - \min\left\{x + \alpha V_{i-1}(y_{m,j}),\ \max\{x, y_{m,j}\} + \alpha V_{i-1}(0)\right\} \left[F_m(y_{m,j}) - F_m(y_{m,j-1})\right] \right| \\
&\leq \max\left\{\alpha \left|V_i(y_{m,1}) - V_{i-1}(y_{m,1})\right|, \alpha \left|V_i(0) - V_{i-1}(0)\right|\right\} F_m(y_{m,1}) \\
&+ \sum_{j=2}^{m} \max\left\{\alpha \left|V_i(y_{m,j}) - V_{i-1}(y_{m,j})\right|, \alpha \left|V_i(0) - V_{i-1}(0)\right|\right\} \left[F_m(y_{m,j}) - F_m(y_{m,j-1})\right] \\
&\leq \alpha \max_{x \in \{0, y_{m,1}, \ldots, y_{m,m}\}} \left|V_i(x) - V_{i-1}(x)\right| F_m(y_{m,1}) \\
&+ \sum_{j=2}^{m} \alpha \max_{x \in \{0, y_{m,1}, \ldots, y_{m,m}\}} \left|V_i(x) - V_{i-1}(x)\right| \left[F_m(y_{m,j}) - F_m(y_{m,j-1})\right] \\
&= \alpha \max_{x \in \{0, y_{m,1}, \ldots, y_{m,m}\}} \left|V_i(x) - V_{i-1}(x)\right| \\
&\leq \alpha \vartheta
\end{aligned}
$$

where the second inequality follows from Lemma 16, and thus

$$\|T_m(V_i) - V_i\| \quad := \quad \sup_{x \in [0,1]} |T_m(V_i)(x) - V_i(x)| \quad \leq \quad \alpha \vartheta$$

□

B.0.3 Approximation $\hat{F}$ by discretization of the range of F.

One shortcoming of the construction of $\hat{F}$ by discretization of the support of F in equal increments described above, is that it does not take into account the distribution of probability in the support. For example, one may obtain a better approximation for the same amount of computational effort by evaluating the integrand where more of the probability is concentrated. The approach described next is based on such considerations.

For any $u \in [0,1]$, let

$$F^{-1}(u) \quad := \quad \inf\{y \in [0,1] \ : \ F(y) \geq u\}$$

Note that F^{-1} is nondecreasing, $F^{-1}(0) = 0$, and $F^{-1}(1) \leq 1$. First, we discretize the range $[0,1]$ of F. For any integer $m' > 0$, let $y_{m',j} \in [F^{-1}((j-1)/m'), F^{-1}(j/m')]$ for $j = 1, \ldots, m'$. For example, one may choose independent uniformly distributed random variables $u_{m',j} \in [(j-1)/m', j/m']$ for $j = 1, \ldots, m'$, and then let $y_{m',j} = F^{-1}(u_{m',j})$. The empirical distribution of such points $y_{m',j}$ will approximate the distribution F. In addition, if $y_{m',j} - F^{-1}((j-1)/m')$ or $F^{-1}(j/m') - y_{m',j}$ is large, we may want to insert a few additional points to better control the error in the approximation of the integral. As before, because f is uniformly continuous on $[0,1]$, for any $\varepsilon > 0$ there is $\delta > 0$ such that $|f(y_2) - f(y_1)| \leq \varepsilon$ for all $y_1, y_2 \in [0,1]$ with $|y_2 - y_1| \leq \delta$. Then choose additional points such that the following hold. The original points $y_{m',j}$ described above together with the new points are denoted with $y_{m,j}$, $j = 1, \ldots, m$, indexed such that $0 \leq y_{m,1} \leq y_{m,2} \leq \cdots \leq y_{m,m} \leq 1$. Consider the interval $[F^{-1}((j-1)/m'), F^{-1}(j/m')]$ for any $j \in \{1, \ldots, m'\}$. Let $y_{m,j'} := \min\{y \in \{y_{m,1}, \ldots, y_{m,m}\} \ : \ F^{-1}((j-1)/m') \leq y\}$, and $y_{m,j''} := \max\{y \in \{y_{m,1}, \ldots, y_{m,m}\} \ : \ y \leq F^{-1}(j/m')\}$. Then it should hold that $y_{m,j'} - F^{-1}((j-1)/m') \leq \delta$, $F^{-1}(j/m') - y_{m,j''} \leq \delta$, and $y_{m,j'''+1} - y_{m,j'''} \leq 2\delta$ for all $j''' \in \{j', \ldots, j''-1\}$. Note that the number of additional points beyond the initial m' points that have to be added to satisfy the conditions above is no more than $\lceil 1/(2\delta) \rceil$, that is, $m \leq m' + \lceil 1/(2\delta) \rceil$. Next, define the sub-probability distributions $F_{m,j}$ for $j = 1, \ldots, m$ as follows: Let $\overline{y}_{m,0} := 0$. If $1/m' \leq F(y_{m,2})$, then let $\overline{y}_{m,1} := F^{-1}(1/m')$ and

$$F_{m,1}(y) \quad := \quad \begin{cases} F(y) & \text{if} \quad y < F^{-1}\left(\frac{1}{m'}\right) \\ \frac{1}{m'} & \text{if} \quad y \geq F^{-1}\left(\frac{1}{m'}\right) \end{cases}$$

else let $\overline{y}_{m,1} := (y_{m,1} + y_{m,2})/2$ and

$$F_{m,1}(y) := \begin{cases} F(y) & \text{if } y < \frac{y_{m,1}+y_{m,2}}{2} \\ F\left(\frac{y_{m,1}+y_{m,2}}{2}\right) & \text{if } y \geq \frac{y_{m,1}+y_{m,2}}{2} \end{cases}$$

For $j = 2, \ldots, m-1$, if $\lfloor m' F_{m,j-1}(1) \rfloor / m' \leq F(y_{m,j+1})$, then let

$\overline{y}_{m,j} := F^{-1}\left((\lfloor m' F_{m,j-1}(1) \rfloor + 1)/m'\right)$ and

$$F_{m,j}(y) := \begin{cases} 0 & \text{if } y < F^{-1}(F_{m,j-1}(1)) \\ F(y) - F_{m,j-1}(1) & \text{if } F^{-1}(F_{m,j-1}(1)) \leq y < F^{-1}\left(\frac{\lfloor m' F_{m,j-1}(1) \rfloor + 1}{m'}\right) \\ \frac{\lfloor m' F_{m,j-1}(1) \rfloor + 1}{m'} - F_{m,j-1}(1) & \text{if } y \geq F^{-1}\left(\frac{\lfloor m' F_{m,j-1}(1) \rfloor + 1}{m'}\right) \end{cases}$$

else let $\overline{y}_{m,j} := (y_{m,j} + y_{m,j+1})/2$ and

$$F_{m,j}(y) := \begin{cases} 0 & \text{if } y < F^{-1}(F_{m,j-1}(1)) \\ F(y) - F_{m,j-1}(1) & \text{if } F^{-1}(F_{m,j-1}(1)) \leq y < \frac{y_{m,j}+y_{m,j+1}}{2} \\ F\left(\frac{y_{m,j}+y_{m,j+1}}{2}\right) - F_{m,j-1}(1) & \text{if } y \geq \frac{y_{m,j}+y_{m,j+1}}{2} \end{cases}$$

Let $\overline{y}_{m,m} := F^{-1}(1)$ and

$$F_{m,m}(y) := \begin{cases} 0 & \text{if } y < F^{-1}(F_{m,m-1}(1)) \\ F(y) - F_{m,m-1}(1) & \text{if } F^{-1}(F_{m,m-1}(1)) \leq y \end{cases}$$

It follows by induction on j that $y_{m,j} \in [\overline{y}_{m,j-1}, \overline{y}_{m,j}]$ for all j, and from the way the points $y_{m,j}$ were chosen it follows that $y_{m,j} - \overline{y}_{m,j-1} \leq \delta$ and $\overline{y}_{m,j}] - y_{m,j} \leq \delta$. Note that $F(y) = \sum_{j=1}^{m} F_{m,j}(y)$, and thus

$$\int_{[0,1]} f(y)\, dF(y) = \sum_{j=1}^{m} \int_{[0,1]} f(y)\, dF_{m,j}(y) = \sum_{j=1}^{m} \int_{[\overline{y}_{m,j-1}, \overline{y}_{m,j}]} f(y)\, dF_{m,j}(y)$$

and that $F_{m,j}(1) \leq 1/m'$ for all j. Define the probability distribution F_m by

$$F_m(y) := \sum_{j=1}^{m} \mathbb{I}_{\{y \geq y_{m,j}\}} F_{m,j}(1)$$

Then $\int_{[0,1]} f(y)\, dF(y)$ is approximated by $\int_{[0,1]} f(y)\, dF_m(y) = \sum_{j=1}^{m} f(y_{m,j}) F_{m,j}(1)$. Then

$$\begin{aligned}
&\left| \int_{[0,1]} f(y)\ dF(y) - \int_{[0,1]} f(y)\ dF_m(y) \right| = \left| \sum_{j=1}^{m} \int_{[0,1]} f(y)\ dF_{m,j}(y) - \sum_{j=1}^{m} f(y_{m,j}) F_{m,j}(1) \right| \\
&= \left| \sum_{j=1}^{m} \left[\int_{[\overline{y}_{m,j-1}, \overline{y}_{m,j}]} f(y)\ dF_{m,j}(y) - \int_{[\overline{y}_{m,j-1}, \overline{y}_{m,j}]} f(y_{m,j})\ dF_{m,j}(y) \right] \right| \\
&\leq \sum_{j=1}^{m} \int_{[\overline{y}_{m,j-1}, \overline{y}_{m,j}]} |f(y) - f(y_{m,j})|\ dF_{m,j}(y) \\
&\leq \sum_{j=1}^{m} \int_{[\overline{y}_{m,j-1}, \overline{y}_{m,j}]} \varepsilon\ dF_{m,j}(y) \\
&= \varepsilon
\end{aligned}$$

B.0.4 Proofs for Section 4.3.1.1

Proof of Lemma 11. Let

$$\begin{aligned}
\Xi_0 &:= \{\xi \in [0, 2\pi)\ :\ \alpha V^*(\xi) < 2 + \alpha V^*(-1)\} \\
\Xi_1 &:= \{\xi \in [0, 2\pi)\ :\ \alpha V^*(\xi) \geq 2 + \alpha V^*(-1)\}
\end{aligned}$$

Then $V^*(-1) = \int_{\Xi_0} \alpha V^*(\xi)\ dF(\xi) + \int_{\Xi_1} [2 + \alpha V^*(-1)]\ dF(\xi)$. Note that for any $\xi \in [0, 2\pi)$,

$$\begin{aligned}
V^*(\xi) &= \int_{[0,2\pi)} \min\left\{2 + \alpha V^*(\xi'),\ 2 + \min\{2, d(\xi, \xi')\} + \alpha V^*(-1)\right\}\ dF(\xi') \\
&\leq \int_{\Xi_0} \left[2 + \alpha V^*(\xi')\right]\ dF(\xi') + \int_{\Xi_1} \left[2 + \min\{2, d(\xi, \xi')\} + \alpha V^*(-1)\right]\ dF(\xi') \\
&\leq 2 + \int_{\Xi_0} \alpha V^*(\xi')\ dF(\xi') + \int_{\Xi_1} [2 + \alpha V^*(-1)]\ dF(\xi') \\
&= 2 + V^*(-1)
\end{aligned}$$

Thus

$$\alpha V^*(\xi) \quad \leq \quad \alpha\,[2 + V^*(-1)] \quad < \quad 2 + \alpha V^*(-1)$$

and thus $V^*(-1) = \int_{[0,2\pi)} \alpha V^*(\xi)\ dF(\xi)$. $\square$

Proof of Proposition 12. For any $x_1, x_2 \in [0, 2\pi)$, it holds that

$$\begin{aligned}
|V^*(x_2) - V^*(x_1)| &= \Bigg| \int_{[0,2\pi)} \min\{2 + \alpha V^*(\xi),\ 2 + \min\{2, d(x_2, \xi)\} + \alpha V^*(-1)\}\ dF(\xi) \\
&- \int_{[0,2\pi)} \min\{2 + \alpha V^*(\xi),\ 2 + \min\{2, d(x_1, \xi)\} + \alpha V^*(-1)\}\ dF(\xi) \Bigg| \\
&\le \int_{[0,2\pi)} |\min\{2 + \alpha V^*(\xi),\ 2 + \min\{2, d(x_2, \xi)\} + \alpha V^*(-1)\} \\
&\qquad - \min\{2 + \alpha V^*(\xi),\ 2 + \min\{2, d(x_1, \xi)\} + \alpha V^*(-1)\}|\ dF(\xi) \\
&\le \int_{[0,2\pi)} \max\{|[2 + \alpha V^*(\xi)] - [2 + \alpha V^*(\xi)]|, \\
&|[2 + \min\{2, d(x_2, \xi)\} + \alpha V^*(-1)] - [2 + \min\{2, d(x_1, \xi)\} + \alpha V^*(-1)]|\}\ dF(\xi) \\
&= \int_{[0,2\pi)} |\min\{2, d(x_2, \xi)\} - \min\{2, d(x_1, \xi)\}|\ dF(\xi) \\
&\le \int_{[0,2\pi)} |d(x_2, \xi) - d(x_1, \xi)|\ dF(\xi) \\
&\le \int_{[0,2\pi)} d(x_1, x_2)\ dF(\xi) \\
&= d(x_1, x_2)
\end{aligned}$$

□

Proof of Proposition 13. Recall that Lemma 11 established that if $x = -1$, then it is optimal to delay service of the new arrival, and thus it is correct that $X^*(\xi)$ does not include -1. Next, consider any $x \in [0, 2\pi)$ and any $\xi \in [0, 2\pi)$. If $d(x, \xi) \le \zeta^*(\xi) := \alpha\,[V^*(\xi) - V^*(-1)]$, then $\min\{2, d(x, \xi)\} + \alpha V^*(-1) = d(x, \xi) + \alpha V^*(-1) \le \alpha V^*(\xi)$, and thus it is optimal to visit the new arrival immediately. If $d(x, \xi) > \zeta^*(\xi)$, then we consider the following 2 cases. First, if $d(x, \xi) \le 2$, then $\min\{2, d(x, \xi)\} + \alpha V^*(-1) = d(x, \xi) + \alpha V^*(-1) > \alpha V^*(\xi)$. Second, if $d(x, \xi) > 2$, then $\min\{2, d(x, \xi)\} + \alpha V^*(-1) = 2 + \alpha V^*(-1) > \alpha V^*(\xi)$, where the last inequality follows from the proof of Lemma 11. In both these cases, it is not optimal to visit the new arrival immediately. □

Proof of Proposition 14. Consider any $x \in [0, 2\pi)$, and any $\xi \in [0, 2\pi)$ such that $\xi \neq x$, $d(x, \xi) = (x - \xi) \bmod 2\pi$, and $\min\{2, d(x, \xi)\} + \alpha V^*(-1) \le \alpha V^*(\xi)$. Recall that it was shown in the proof of Lemma 11 that $\alpha V^*(\xi) < 2 + \alpha V^*(-1)$ for all $\xi \in [0, 2\pi)$, and thus it follows that $d(x,\xi) < 2$ and $\min\{2, d(x,\xi)\} + \alpha V^*(-1) = d(x,\xi) + \alpha V^*(-1) \leq \alpha V^*(\xi)$.

Consider any $\xi' \in [0, 2\pi)$ such that $d(x,\xi') = (x - \xi') \bmod 2\pi < (x - \xi) \bmod 2\pi = d(x,\xi)$. It follows that $\min\{2, d(x,\xi')\} = d(x,\xi')$. We show by contradiction that $(\xi' - \xi) \bmod 2\pi \leq \pi$ and thus $d(\xi',\xi) = (\xi' - \xi) \bmod 2\pi$. Suppose that $(\xi' - \xi) \bmod 2\pi > \pi$, then $0 < (\xi - \xi') \bmod 2\pi < \pi$. Then $d(x,\xi') = (x - \xi') \bmod 2\pi = (x - \xi + \xi - \xi') \bmod 2\pi = (x - \xi) \bmod 2\pi + (\xi - \xi') \bmod 2\pi > (x - \xi) \bmod 2\pi$, which contradicts the assumption that $(x - \xi') \bmod 2\pi < (x - \xi) \bmod 2\pi$. The third equality follows from the assumptions that $d(x,\xi) = (x - \xi) \bmod 2\pi \leq \pi$ and $(\xi - \xi') \bmod 2\pi < \pi$, and thus $(x - \xi) \bmod 2\pi + (\xi - \xi') \bmod 2\pi < 2\pi$. Thus, $(x - \xi) \bmod 2\pi = (x - \xi' + \xi' - \xi) \bmod 2\pi = (x - \xi') \bmod 2\pi + (\xi' - \xi) \bmod 2\pi$, where the last equality follows from the assumption that $d(x,\xi') = (x - \xi') \bmod 2\pi \leq \pi$, the result shown above that $(\xi' - \xi) \bmod 2\pi \leq \pi$, and the assumption that $\xi \neq x$. Hence, $(x - \xi') \bmod 2\pi - (x - \xi) \bmod 2\pi = -(\xi' - \xi) \bmod 2\pi$.

Next, note that it follows from Proposition 12 that $|V^*(\xi) - V^*(\xi')| \leq d(\xi',\xi)$, and thus

$$\begin{aligned}
\alpha \left[V^*(\xi) - V^*(\xi')\right] &\leq \alpha \left|V^*(\xi) - V^*(\xi')\right| \\
&\leq \left|V^*(\xi) - V^*(\xi')\right| \\
&\leq d(\xi',\xi) \quad = \quad \{(\xi' - \xi) \bmod 2\pi\}
\end{aligned}$$

Hence,

$$\begin{aligned}
& \left[\min\{2, d(x,\xi')\} - \alpha V^*(\xi')\right] - \left[\min\{2, d(x,\xi)\} - \alpha V^*(\xi)\right] \\
= \; & \left[(x - \xi') \bmod 2\pi - \alpha V^*(\xi')\right] - \left[(x - \xi) \bmod 2\pi - \alpha V^*(\xi)\right] \\
= \; & \alpha \left[V^*(\xi) - V^*(\xi')\right] - (\xi' - \xi) \bmod 2\pi\} \quad \leq \quad 0
\end{aligned}$$

Therefore,

$$\min\{2, d(x,\xi')\} + \alpha V^*(-1) - \alpha V^*(\xi') \quad \leq \quad \min\{2, d(x,\xi)\} + \alpha V^*(-1) - \alpha V^*(\xi) \quad \leq \quad 0$$

that is, $\min\{2, d(x,\xi')\} + \alpha V^*(-1) \leq \alpha V^*(\xi')$.

Recall the observation above that $\min\{2, d(x,\xi)\} + \alpha V^*(-1) \leq \alpha V^*(\xi)$ implies that $d(x,\xi) < 2$. Hence, if $\{(x - \xi) \bmod 2\pi\} + \alpha V^*(-1) \leq \alpha V^*(\xi)$, then $(x - \xi) \bmod 2\pi < 2 < \pi$, and thus $d(x,\xi) = (x-\xi) \bmod 2\pi$. Thus the definition of z_-^* can be simplified to

$$z_-^*(x) \quad = \quad \sup\left\{(x-\xi) \bmod 2\pi \ : \ \xi \in [0,2\pi), (x-\xi) \bmod 2\pi + \alpha V^*(-1) \leq \alpha V^*(\xi)\right\}$$

Since $V^*(-1) \leq V^*(\xi)$ for all ξ, it follows that $z_-^*(x) \geq 0$ for all x. Also, it follows from the remarks above that $z_-^*(x) \leq 2$ for all x. In fact, it follows from the continuity of $(x-\xi)$ mod 2π and $V^*(\xi)$ in ξ that there exists a $\bar{\xi} \in [0,2\pi)$ such that $z_-^*(x) = (x-\bar{\xi}) \bmod 2\pi$ and $(x-\bar{\xi}) \bmod 2\pi + \alpha V^*(-1) = \alpha V^*(\bar{\xi})$, and thus $z_-^*(x) < 2$ for all x. Next, it also follows from the results above that if $(x-\xi') \bmod 2\pi \leq z_-^*(x)$ then $d(x,\xi') = (x-\xi') \bmod 2\pi$ and $\min\{2, d(x,\xi')\} + \alpha V^*(-1) \leq \alpha V^*(\xi')$, and it follows from the definition of z_-^* that if $(x-\xi')$ mod $2\pi > z_-^*(x)$ then $\alpha V^*(\xi') < \{(x-\xi') \bmod 2\pi\} + \alpha V^*(-1)$.

Similarly,

$$z_+^*(x) \quad = \quad \sup\left\{(\xi-x) \bmod 2\pi \ : \ \xi \in [0,2\pi), (\xi-x) \bmod 2\pi + \alpha V^*(-1) \leq \alpha V^*(\xi)\right\}$$

and if $(\xi'-x) \bmod 2\pi \leq z_+^*(x)$ then $d(x,\xi') = (\xi'-x) \bmod 2\pi$ and $\min\{2, d(x,\xi')\} + \alpha V^*(-1) \leq \alpha V^*(\xi')$, and it follows from the definition of z_+^* that if $(\xi'-x) \bmod 2\pi > z_+^*(x)$ then $\alpha V^*(\xi') < \{(\xi'-x) \bmod 2\pi\} + \alpha V^*(-1)$. Finally, it follows from the results above that if $(\xi'-x) \bmod 2\pi > z_+^*(x)$ and $(x-\xi') \bmod 2\pi > z_-^*(x)$ then $\alpha V^*(\xi') < \{(\xi'-x) \bmod 2\pi\} + \alpha V^*(-1)$ and $\alpha V^*(\xi') < \{(x-\xi') \bmod 2\pi\} + \alpha V^*(-1)$ and $\alpha V^*(\xi') < 2 + \alpha V^*(-1)$, and thus $\alpha V^*(\xi') < \min\{2, d(x,\xi')\} + \alpha V^*(-1)$. □

B.0.5 Proofs for Section 4.4.1.1

$$\begin{aligned}
V^*(x) \quad &= \quad \mathbb{E}_F\left[\min_{u\in\{0,1\}} \{c(x,\xi,u) + \alpha V^*(f(\xi,u))\}\right] \\
&= \quad \begin{cases}
\int_{[0,1]\times[0,2\pi)} \min\left\{\alpha V^*(\xi),\ 2r_\xi + \alpha V^*(-1)\right\}\ dF(\xi) & \text{if} \quad x = -1 \\
\int_{[0,1]\times[0,2\pi)} \min\left\{2r_x + \alpha V^*(\xi),\ 2r_x + d'(x,\xi) + \alpha V^*(-1)\right\}\ dF(\xi) & \\
\text{if } x \in [0,1]\times[0,2\pi) &
\end{cases}
\end{aligned} \tag{19}$$

Proof of Lemma 15. Let

$$\begin{aligned}
\Xi_0 \quad &:= \quad \{\xi \in [0,1]\times[0,2\pi) \ : \ \alpha V^*(\xi) \leq 2r_\xi + \alpha V^*(-1)\} \\
\Xi_1 \quad &:= \quad \{\xi \in [0,1]\times[0,2\pi) \ : \ \alpha V^*(\xi) > 2r_\xi + \alpha V^*(-1)\}
\end{aligned}$$

Then $V^*(-1) = \int_{\Xi_0} \alpha V^*(\xi)\ dF(\xi) + \int_{\Xi_1} [2r_\xi + \alpha V^*(-1)]\ dF(\xi)$. Note that for any $\xi \in [0,1] \times [0, 2\pi)$,

$$\begin{aligned}
V^*(\xi) &= \int_{[0,1]\times[0,2\pi)} \min\left\{2r_\xi + \alpha V^*(\xi'),\ 2r_\xi + d'(\xi,\xi') + \alpha V^*(-1)\right\}\ dF(\xi') \\
&\leq \int_{\Xi_0} \left[2r_\xi + \alpha V^*(\xi')\right]\ dF(\xi') + \int_{\Xi_1} \left[2r_\xi + d'(\xi,\xi') + \alpha V^*(-1)\right]\ dF(\xi') \\
&\leq 2r_\xi + \int_{\Xi_0} \alpha V^*(\xi')\ dF(\xi') + \int_{\Xi_1} \left[2r_{\xi'} + \alpha V^*(-1)\right]\ dF(\xi') \\
&= 2r_\xi + V^*(-1)
\end{aligned}$$

Thus

$$\alpha V^*(\xi) \quad \leq \quad \alpha\left[2r_\xi + V^*(-1)\right] \quad \leq \quad 2r_\xi + \alpha V^*(-1)$$

and thus $V^*(-1) = \int_{[0,1]\times[0,2\pi)} \alpha V^*(\xi)\ dF(\xi)$. □

REFERENCES

[1] Ak, A. and Erera, A. L., "A paired-vehicle recourse strategy for the vehicle-routing problem with stochastic demands," *Transportation Science*, vol. 41, no. 2, pp. 222–237, 2007.

[2] Angelelli, E., Speranza, M. G., and Savelsbergh, M. W. P., "Competitive analysis for dynamic multiperiod uncapacitated routing problems," *Networks*, vol. 49, no. 4, pp. 308–317, 2007.

[3] Bartholdi, J. and Platzman, L., "An n logn planar traveling salesman heuristic based on space-filling curves," *Operations Research Letters*, vol. 1, pp. 121–125, 1982.

[4] Bastian, C. and Kan, A., "The stochastic vehicle-routing problem revisited," *European Journal of Operational Research*, vol. 56, no. 3, pp. 407–412, 1992.

[5] Benders, J., "Partitioning procedures for solving mixed-variables programming problems," *Numerische Mathematik*, vol. 4, pp. 238–252, 1962.

[6] Bent, R. W. and Van Hentenryck, P., "Scenario-based planning for partially dynamic vehicle routing with stochastic customers," *Operations Research*, vol. 52, no. 6, pp. 977–987, 2004.

[7] Benton, W. C. and Rossetti, M. D., "The vehicle scheduling problem with intermittent customer demands," *Computers and Operations Research*, vol. 19, no. 6, pp. 521–531, 1992.

[8] Bertsimas, D. J., *Probabilistic combinatorial optimization problems.* Phd thesis, Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, 1988.

[9] Bertsimas, D. J., "A vehicle-routing problem with stochastic demand," *Operations Research*, vol. 40, no. 3, pp. 574–586, 1992.

[10] Bertsimas, D. J., Chervi, P., and Peterson, M., "Computational approaches to stochastic vehicle routing problems," *Transportation Science*, vol. 29, no. 4, pp. 342–352, 1995.

[11] Bertsimas, D. J. and Howell, L., "Further results on the probabilistic traveling salesman problem," *European Journal of Operational Research*, vol. 65, pp. 68–85, 1993.

[12] Bertsimas, D. J., Jaillet, P., and Odoni, A. R., "A priori optimization," *Operations Research*, vol. 38, no. 6, pp. 1019–1033, 1990.

[13] Bertsimas, D. J. and SimchiLevi, D., "A new generation of vehicle routing research: Robust algorithms, addressing uncertainty," *Operations Research*, vol. 44, no. 2, pp. 286–304, 1996.

[14] Bertsimas, D. J. and Vanryzin, G., "A stochastic and dynamic vehicle-routing problem in the euclidean plane," *Operations Research*, vol. 39, no. 4, pp. 601–615, 1991.

[15] Bramel, J. and Simchi-Levi, D., "A location based heuristic for general routing problems," *Operations Research*, vol. 43, pp. 649–660, 1995.

[16] Braysy, O. and Gendreau, M., "Vehicle routing problem with time windows, part i: route construction and local search algorithms," *Transportation Science*, vol. 39, pp. 104–118, 2005.

[17] Campbell, A. M. and Thomas, B. W., "Probabilistic traveling salesman problem with deadlines," *Transportation Science*, vol. 42, no. 1, pp. 1–21, 2008.

[18] Carraway, R. L., Morin, T., and Moskowitz, H., "Generalized dynamic programming for stochastic combinatorial optimization," *Operations Research*, vol. 37, pp. 819–829, 1989.

[19] Charnes, A. and Cooper, W., "Chance-constrained programming," *Management Science*, vol. 6, pp. 73–79, 1959.

[20] Clarke, G. and Wright, J. W., "Scheduling of vehicles from central depot to number of delivery points," *Operations Research*, vol. 12, no. 4, pp. 568–581, 1964.

[21] Cordeau, J.-F., Desaulniers, G., Desrosiers, J., Solomon, M., and Soumis, F., "Vrp with time windows," *SIAM Monographs on Discrete Mathematics and Applications, SIAM*, pp. 157–193, 2002.

[22] Dror, M., Laporte, G., and Louveaux, F., "Vehicle routing with stochastic demands and restricted failures," *Zeitschrift fur Operations Research*, vol. 37, pp. 273–283, 1993.

[23] Dror, M., Laporte, G., and Trudeau, P., "Vehicle-routing with stochastic demands - properties and solution frameworks," *Transportation Science*, vol. 23, no. 3, pp. 166–176, 1989.

[24] Dror, M. and Trudeau, P., "Stochastic vehicle-routing with modified savings algorithm," *European Journal of Operational Research*, vol. 23, no. 2, pp. 228–235, 1986.

[25] Erera, A. L., Savelsbergh, M., and Uyar, E., "Fixed routes with backup vehicles for stochastic vehicle routing problems with time constraints," *submitted to Networks*, 2008.

[26] Fisher, M. and Jaikumar, R., "A generalized assignment heuristic," *Networks*, vol. 11, pp. 109–124, 1981.

[27] Gabow, H. N., "Efficient implementation of edmonds algorithm for maximum matching on graphs," *Journal of the Acm*, vol. 23, no. 2, pp. 221–234, 1976.

[28] Gabow, H. N., "Data structures for weighted matching and nearest common ancestors with linking.," *Proceedings of the first annual ACM-SIAM symposium on Discrete Algorithms*, pp. 434–443, 1990.

[29] Garey, M., Johnson, D., and Stockmeyer, L., "Some simplified np-complete problems," *Annual ACM Symposium on Theory of Computing, Proceedings of the sixth annual ACM symposium on Theory of computing*, pp. 47–63, 1974.

[30] Gendreau, M., Guertin, F., Potvin, J. Y., and Taillard, E., "Parallel tabu search for real-time vehicle routing and dispatching," *Transportation Science*, vol. 33, no. 4, pp. 381–390, 1999.

[31] Gendreau, M., Laporte, G., and Seguin, R., "An exact algorithm for the vehicle-routing problem with stochastic demands and customers," *Transportation Science*, vol. 29, no. 2, pp. 143–155, 1995.

[32] Gendreau, M., Laporte, G., and Seguin, R., "Stochastic vehicle routing," *European Journal of Operational Research*, vol. 88, no. 1, pp. 3–12, 1996.

[33] Gendreau, M., Laporte, G., and Seguin, R., "A tabu search heuristic for the vehicle routing problem with stochastic demands and customers," *Operations Research*, vol. 44, no. 3, pp. 469–477, 1996.

[34] Haughton, M. A., "The performance of route modification and demand stabilization strategies in stochastic vehicle routing," *Transportation Research Part B-Methodological*, vol. 32, no. 8, pp. 551–566, 1998.

[35] Haughton, M. A., "Quantifying the benefits of route reoptimisation under stochastic customer demands," *Journal of the Operational Research Society*, vol. 51, no. 3, pp. 320–332, 2000.

[36] Hjorring, C. and Holt, J., "New optimality cuts for a single-vehicle stochastic routing problem," *Annals of Operations Research*, vol. 86, pp. 569–584, 1999.

[37] Jaillet, P., *Probabilistic traveling salesman problem.* PhD thesis, Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA, 1985.

[38] Jaillet, P., "Stochastic routing problems," *Stochastics in Combinatorial Optimization*, pp. 197–213, 1987.

[39] Jaillet, P., "Apriori solution of a traveling salesman problem in which a random subset of the customers are visited," *Operations Research*, vol. 36, no. 6, pp. 929–936, 1988.

[40] Jaillet, P. and Odoni, A. R., "The probabilistic vehicle routing problem," *Vehicle Routing: Methods and Studies*, pp. 293–318, 1988.

[41] Jezequel, A., *Probabilistic vehicle routing problems.* Msc thesis, Department of Civil Engineering, Massachusetts Institute of Technology, Cambridge, MA, 1985.

[42] Kao, E., "A preference order dynamic program for a stochastic traveling salesman problem," *Operations Research*, vol. 26, pp. 1033–1045, 1978.

[43] Kenyon, A. S. and Morton, D. P., "Stochastic vehicle routing with random travel times," *Transportation Science*, vol. 37, no. 1, pp. 69–82, 2003.

[44] Lambert, V., Laporte, G., and Louveaux, F., "Designing collection routes through bank branches," *Computers and Operations Research*, vol. 20, no. 7, pp. 783–791, 1993.

[45] Laporte, G., Louveaux, F., and Mercure, H., "The vehicle-routing problem with stochastic travel-times," *Transportation Science*, vol. 26, no. 3, pp. 161–170, 1992.

[46] Laporte, G., Louveaux, F., and van Hamme, L., "An integer l-shaped algorithm for the capacitated vehicle routing problem with stochastic demands," *Operations Research*, vol. 50, no. 3, pp. 415–423, 2002.

[47] Laporte, G. and Louveaux, F. V., "Formulations and bounds for the stochastic capacitated vehicle routing problem with uncertain supplies," *Economic Decision Making:Games, Econometrics and Optimisation*, pp. 443–455, 1990.

[48] Laporte, G. and Louveaux, F. V., "The integer l-shaped method for stochastic integer programs with complete recourse," *Operations Research Letters*, vol. 13, no. 3, pp. 133–142, 1993.

[49] Laporte, G., Louveaux, F. V., and Mercure, H., "A-priori optimization of the probabilistic traveling salesman problem," *Operations Research*, vol. 42, no. 3, pp. 543–549, 1994.

[50] Larsen, A., *The dynamic vehicle routing problem.* Phd thesis, Technical University of Denmark, Kongens, Lyngby, Denmark, 2000.

[51] Leipala, T., "Solutions of stochastic traveling salesman problems," *European Journal of Operational Research*, vol. 2, no. 4, pp. 291–297, 1978.

[52] Morales, J., *Planning Robust Freight Transportation Operations.* Phd thesis, Industrial and Systems Engineering, Georgia Institute of Technology, Atlatna, GA, 2006.

[53] Psaraftis, H. N., "Dynamic vehicle routing: Status and prospects," *Annals of Operations Research*, vol. 61, pp. 143–164, 1995.

[54] Rossi, F. and Gavioli, I., "Aspects of heuristic methods in the "probabilistic traveling salesman problem"," *Stochastics in Combinatorial Optimization*, pp. 214–227, 1987.

[55] Savelsbergh, M. and Goetschalckx, M., "A comparison of the efficiency of fixed versus variable vehicle routes," *Journal of Business Logistics*, vol. 16, pp. 163–187, 1995.

[56] Secomandi, N., *Exact and heuristic dynamic programming algorithms for the vehicle routing problem with stochastic demands.* Phd thesis, Faculty of the College of Business Administration, University of Houston, Houston, TX, 1998.

[57] Secomandi, N., "Comparing neuro-dynamic programming algorithms for the vehicle routing problem with stochastic demands," *Computers and Operations Research*, vol. 27, no. 11-12, pp. 1201–1225, 2000.

[58] Secomandi, N., "A rollout policy for the vehicle routing problem with stochastic demands," *Operations Research*, vol. 49, pp. 796–802, 2001.

[59] Secomandi, N., "Analysis of a rollout approach to sequencing problems with stochastic routing applications," *Journal of Heuristics*, vol. 9, no. 4, pp. 321–352, 2003.

[60] Seguin, R., *Problemes stochastiques de tournees de vehicules.* Phd thesis, Departement d'informatique et de recherche operationelle, Universite de Montreal, Montreal, Canada, 1994.

[61] Sleator, D. and Tarjan, R. E., "Amortized efficiency of list update and paging rules," *Communications of ACM*, vol. 28, pp. 202–208, 1985.

[62] Sniedovich, M., "Analysis of a preference order traveling salesman problem," *Operations Research*, vol. 29, pp. 1234–1237, 1981.

[63] Solomon, M. M., "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Operations Research*, vol. 35, pp. 254–265, 1987.

[64] Stewart, W. R. and Golden, B. L., "Stochastic vehicle-routing - a comprehensive approach," *European Journal of Operational Research*, vol. 14, no. 4, pp. 371–385, 1983.

[65] Tillman, F., "The multiple terminal delivery problem with probabilistic demands," *Transportation Science*, vol. 31, pp. 192–204, 1969.

[66] Van Hentenryck, P. and Bent, R., *Online Stochastic Combinatorial Optimization.* The MIT Press, 2006.

[67] Verweij, B., Ahmed, S., Kleywegt, A. J., Nemhauser, G., and Shapiro, A., "The sample average approximation method applied to stochastic routing problems: A computational study," *Computational Optimization and Applications*, vol. 24, no. 2-3, pp. 289–333, 2003.

[68] Yang, W. H., Mathur, K., and Ballou, R. H., "Stochastic vehicle routing problem with restocking," *Transportation Science*, vol. 34, no. 1, pp. 99–112, 2000.